EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Adversarial attacks on fingerprint liveness detection

Jianwei Fei[1], Zhihua Xia[1*], Peipeng Yu[1] and Fengjun Xiao[2]

Abstract

Deep neural networks are vulnerable to adversarial samples, posing potential threats to the applications deployed with deep learning models in practical conditions. A typical example is the fingerprint liveness detection module in fingerprint authentication systems. Inspired by great progress of deep learning, deep networks-based fingerprint liveness detection algorithms spring up and dominate the field. Thus, we investigate the feasibility of deceiving state-of-the-art deep networks-based fingerprint liveness detection schemes by leveraging this property in this paper. Extensive evaluations are made with three existing adversarial methods: FGSM, MI-FGSM, and Deepfool. We also proposed an adversarial attack method that enhances the robustness of adversarial fingerprint images to various transformations like rotations and flip. We demonstrate these outstanding schemes are likely to classify fake fingerprints as live fingerprints by adding tiny perturbations, even without internal details of their used model. The experimental results reveal a big loophole and threats for these schemes from a view of security, and enough attention is urgently needed to be paid on anti-adversarial not only in fingerprint liveness detection but also in all deep learning applications.

Keywords: Deep learning, Fingerprint liveness detection, Adversarial attacks

1 Introduction

The rapid growth in deep learning and in particular convolutional neural networks (CNNs) brings new solutions to many problems in computer vision, big data [1], and security [2]. These breakthroughs are gradually being put into use of various practical applications like face identification [3–5], pedestrian detection [6, 7], and unmanned vehicles [8, 9]. While deep networks have seen phenomenal success in many domains, Szegedy et al. [10] first demonstrated that through intentionally adding certain tiny perturbations, an image remains indistinguishable to original image but networks probably misclassify it as other classes instead of the original prediction. This is called *adversarial attack* and the perturbed image is the namely *adversarial sample*. Part of their results is shown in Fig. 1. It is interesting that we notice the perturbation images show some similarity with the encrypted images [12–16], but the former are magnified noise while the latter are sophisticated designed encrypted files. Recent researchers have created serval methods to craft adversarial samples which vary greatly in terms of perturbation degree, number of perturbed pixels, and computation complexity.

There are serval sorting criterions of adversarial attacks concerning the level that attackers are in the know of target models or whether the misclassified label is specified. Generating adversarial samples with the architecture and parameters of the target model is referred to as *white-box attack* while *black-box attack* without them. For an image, if not only the attack is required to be successful, but also the adversarial sample generated is required to classified to a specific class, it is called *targeted attack* and otherwise *untargeted attack*. Generating adversarial samples is a constrained optimization problem. Given a clean image and a fixed classifier that originally makes correct classification, our goal is to make the classifier misclassify the clean image. Note that the prediction results can be regarded as a function of the clean image about the classifier of which the parameters are fixed. Thus, general adversarial attack methods computing gradients of the clean image about the classifier to

* Correspondence: xia_zhihua@163.com
[1]Jiangsu Engineering Center of Network Monitoring, Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
Full list of author information is available at the end of the article

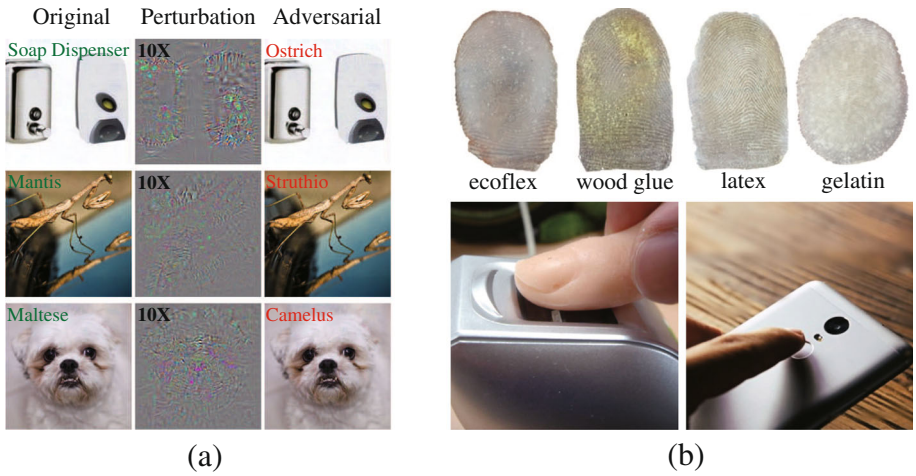

Fig. 1 **a** Adversarial samples generated with AlexNet [11] in [10]. The left column shows the correctly predicted samples, and the middle column is the magnified value of perturbations. The adversarial samples and target labels are shown in the rightmost column. **b** Fake fingerprints made from different materials and cheating authentication system or unlocking smartphones with them

make the prediction deviate from the original result, and modify the clean image accordingly.

Since Szegedy et al. [10] explored this property, and with many efficient, robust attack methods being crafted continuously, a potential security threat for practical deep learning applications came into view. For instance, face recognition systems using CNNs also show vulnerability against adversarial samples [17–19]. Such biometric information is always used with sensitive purposes or scenarios requiring high security, especially fingerprint due to its uniqueness varies individuals. Considering this, we extend similar work on another application referred to as fingerprint liveness detection in this paper, notice that we are the first introducing adversarial attacks into this area to our knowledge. The fingerprint liveness detection module is always deployed in fingerprint authentication systems. This technology aims to distinguish whether the fingerprint is an alive part of a person or a fake one forged with silicone, etc. It is in general divided into hardware- and software-based approaches depending on whether additional sensors are required. The latter can be easily developed into most systems therefore received more attention, and it can be further classified as feature- and deep learning-based. Among them, deep learning-based solutions caused a rising interest in recent years thanks to the rising of deep learning. Although they reached much more outstanding performance than feature-based solutions, the vulnerable property of CNNs leaves a potential risk. That is, the correctly classified fake fingerprint can pass through the detection module by presenting its adversarial sample. Even though attackers cannot successfully cheat fingerprint recognition system with fake fingerprint, they may still against the system by supplying an adversarial fingerprint image. In this paper, we thoroughly evaluate the robustness of several state-of-the-art fingerprint liveness detection models by both white-box and black-box attacks in various settings and demonstrate the vulnerability of these models in this setting.

In our paper, we successfully attack deep learning-based fingerprint liveness detection methods, including the-state-of-the-art one by adversarial attack technology. Sufficient experiments show that once these methods are open source, for almost any fingerprint, the malicious can make its adversarial sample to pose as an alive one and cheat the detection algorithms. Our work also shows even if the details of these detection algorithms are unknown, there is still a definite possibility to realize this attack. We also propose an enhanced adversarial attack method to generate adversarial samples that are more robust to various transformations and achieve a higher attack success rate compared to other advanced methods.

2 Related work

In this section, we will review the development of adversarial attack methods and deep learning-based fingerprint liveness detection models. On the basis of current knowledge, deep neural networks achieve high performance on tasks in computer vision and natural language processing because they can characterize arbitrary continuous function with an incalculable number of cascaded nonlinear steps. But as the result is automatically computed by backpropagation via supervised learning, it can be difficult to interpret and can have counterintuitive properties. And with deep neural networks' increasing usage in the physical world, these properties may be used for malicious behavior.

Szegedy et al. [10] first revealed that adding a certain hardly perceptible perturbation which increasing the prediction error could cause networks to misclassify an image. They also found this property is not affected by the structure and dimensionality of networks or data distribution, and even more, the same perturbation could cause misclassifications on different networks with the same original input image. They proposed an equation that searches the smallest perturbation added to cause misclassification:

$$\begin{aligned} &\text{minimize } \|p\|_2 \text{ s.t. } f(\mathrm{X}_c + p) \\ &\quad = y_{\text{target}}; \mathrm{X}_c + p \in [0, 1] \end{aligned} \tag{1}$$

This is a hard problem, hence the author approximated it using a box-constrained L-BFGS [20] and it turns into a convex optimization process. This is completing by searching the minimum $c > 0$ where the minimizer p of the following problem satisfies $f(\mathrm{X}_c + p) = y_{\text{target}}$:

$$\begin{aligned} \text{minimize } c \mid p \mid + \text{Loss}_f\left(\mathrm{X}_c + p, y_{\text{target}}\right). \text{s.t } \mathrm{X}_c \\ + p \in [0, 1] \end{aligned} \tag{2}$$

As shown in Fig. 1a, by solving this optimization problem, we could compute the perturbations to which a clean image that could successfully fool a model should be added, but the adversarial images and original images are hardly distinguishable to human. It was also observed that a considerable number of adversarial examples will be misclassified by different networks as well, namely, *cross model generalization*. These astonishing discoveries aroused strong interest of researchers in adversarial attacks of computer vision and gave birth to related competitions [21, 22].

In ICLR 2015, Goodfellow et al. [23] proposed a method referred to as *Fast Gradient Sign Method* (FGSM) to efficiently compute the perturbation by solving the following problem:

$$p = \varepsilon \ \text{sign}\left(\nabla J\left(\theta, \mathrm{X}_c, y_{\text{target}}\right)\right) \tag{3}$$

where $\nabla J(\ldots)$ computes the gradient of the cost function around parameters of the model w.r.t. X_c and ε notes a small coefficient that restricts the infinite norm of the perturbation. They successfully caused a misclassification rate of 99.9% on a shallow softmax classifier trained on MNIST while $\varepsilon = 0.25$ and 87.15% on a convolutional maxout network trained on CIFAR-10 while $\varepsilon = 0.1$. Miyato et al. [24] then normalized the computed perturbation with L_2-norm on this basis. FGSM and its varietas are classic one-shot method that generates an adversarial sample with one step only. Later in 2017, Kurakin et al. [25] developed an iterative method that takes multiple steps increasing the loss function namely *Basic Iterative Method* (BIM). Their approach exceedingly reduces the size of perturbation for generating an adversarial sample and shows a serious threat to deep architecture models such as Inception-v3 [26]. Similarly, Moosavi-Dezfooli et al. [27] proposed Deepfool that also computes the minimum perturbation iteratively. This algorithm disturbs the image with a small vector, pushing the clean image confined in the decision boundary out of the boundary step by step until the misclassification occurs. Dong et al. [28] introduced momentum into FGSM, in their approach, not only the current gradient is computed during every iteration but also the gradient of the last iteration is added, and a decay factor is used to control the influence of the previous gradient. This *Momentum Iterative Method* (MIM) greatly improves cross model generalization and black-box attack success rate, their team won the first prize in NIPS 2017 Non-targeted Adversarial Attack and Targeted Adversarial Attack competitions [21]. The above methods all compute the perturbation by solving a gradient related problem, usually requiring direct access to target models. To realize a more robust black-box attack, Su et al. [29] proposed *One Pixel Attack* that searches the perturbation by differential evolution that causes misclassification with the highest confidence instead of computing the gradient. This method made no restraint of perturbation size, meanwhile, it limits the number of perturbed pixels.

With the development of adversarial attack technology, some scholars began to conduct research on attacking real-world systems embedded with deep learning algorithms. Kurakin et al. [25] first proved that the threat of adversarial attack also exists in the real world. They printed adversarial images and took snapshots from smartphones. Results show that even through captured by camera, a relatively large part of adversarial images are misclassified as well. Kevin et al. [30] designed Robust Physical Perturbations (RP2) which only perturbs the target objects in physical world such as guideposts and keeps the background unchanged. For instance, sticking several black and white stickers on a stop sign according to RP2's result could prevent YOLO and Faster-RCNN from detecting it correctly. Bose et al. [31] also successfully attacks Faster-RCNN with adversarial examples that crafted from their proposed adversarial generator network by solving a constrained optimization problem.

In addition to face location, another key problem in face recognition is liveness detection. Biometrics like faces are usually applied in systems with high-security requirements, thus the systems are always accompanied by liveness detection module to detect whether a captured face image is alive or from photos. We note that fingerprint identification systems also require liveness detection to distinguish live fingers from fake ones [32], and with more and more fingerprint liveness detection algorithms based

on deep learning are developed, the adversarial attack has risen a potential risk in this domain as well. To our knowledge, Nogueira et al. [33] first detected fake fingerprint using CNNs, later in [34], they fine-tuned the fully connected layer of VGG and Alexnet with fingerprint datasets, leaving previous convolutional and pooling layers unchanged. This work has reached astonishing performance compared to feature-based approaches in fingerprint liveness detection. Chugh et al. [35] cut fingerprint patches centered on pre-extracted minutiaes and trained them with Mobilenet-v1. Their results are state-of-the-art as we got on with this work. In the literature, Kim et al. [36] proposed a detection algorithm based on deep belief network (DBN) that is constructed layer by layer using restricted Boltzmann machines (RBM). Nguyen et al. [37] regarded the fingerprint as a global texture feature and designed an end-to-end model following this idea. Their experimental results show that networks designed to combine the inherent characteristics of fingerprints can achieve better performance. Pala et al. [38] constructed a triple dataset to train their network. A triple set consists of a fingerprint to be detected, a fingerprint of the same class as it and a fingerprint of the other class. This data structure could make a constraint to minimize within-class distance and maximize between-class class distance is as large as possible. It is noteworthy that all these methods mentioned are based on CNN, and achieved very competitive performances.

3 Methods

3.1 Networks to be attacked

3.1.1 VGG19 and Alexnet-based method

In this section, we will briefly introduce the target networks we attempt to attack, including specific structure and training processes. Before we conduct adversarial attacks on the state-of-the-art fingerprint liveness detection networks, a pre-evaluation would be carried on [34], the finetuned VGG and Alexnet. This is because the way that finetuning classical models for new tasks is widely used, though these models are a bit out of date, they stood the test of time and from which more advanced models derive. Equally thorough experiments will also be carried on [35]. According to Nogueira's method in [34], both models are finetuned with stochastic gradient descent (SGD) while batch size is 5, momentum [39] is 0.9, and the learning rate is fixed at 1E−6.

In these two models, both outputs fully connected layers are replaced by 2 units which were 1024 in original networks as shown in Fig. 2. For keeping a concise but intuitive impression, the size of these feature maps is not prorated and pooling operations are represented by shrinkage of it. In pre-process, the training set is augmented by the implementation similar to the one in [11], patches with 80% of each dimension of the original images are cut for each fingerprint image, thus we totally obtain five patches from four corners and center and create horizontal reflection version of them. The whole training set is therefore 10 times larger than the original edition. During the testing phase, the testing set adopts the same approach and fuse the 10 patch's prediction as to the final classification results for a single fingerprint image.

3.1.2 Mobilenet-v1-based method

Chugh's method also utilizes an existing structure called Mobilenet-v1 but train it from scratch. The last layer is replaced by a 2-unit softmax layer as well. In pre-process, they extracted minutiaes using the algorithm from [40] for a fingerprint image, a minutiae is a key point in fingerprint images, for instance, ridge ending, short or independent ridge and the circle in the ridge pattern. A minutiae object returns x, y coordinate and its direction. Then cut out patches centered on the coordinates, and align the patches according to the directions in order to cut out smaller ones. All the patches are used

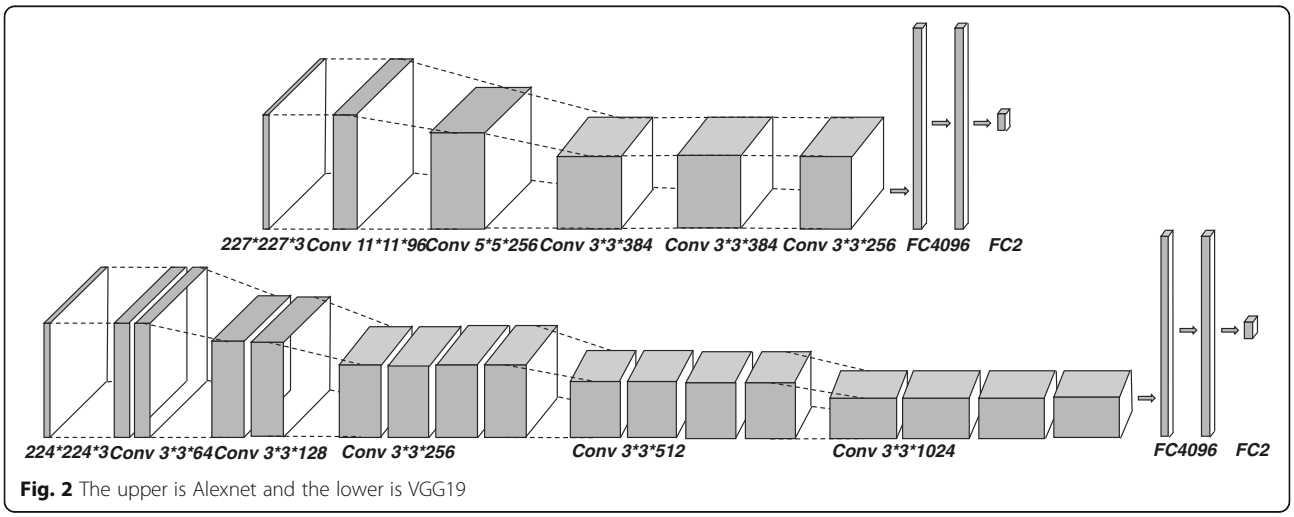

Fig. 2 The upper is Alexnet and the lower is VGG19

to train a Mobilenet-v1, the result is a fusion of all the patches' scores Fig. 3. This series of operations is on the basis that a fingerprint image has large blank areas surrounding the ridge region, directly resizing these images would lead to a serious discriminatory information loss. The noise involved in the fingerprint forgery process provides salient cues to distinguish a spoof fingerprint from live fingerprints, thus patches centered at minutiaes could maximize this difference. This is the best fingerprint liveness detection method at present to our knowledge.

3.2 Methods to generate samples

In this paper, we totally compared four algorithms regarding success rate, visual impact, and robust to transformations. FGSM is the first basic adversarial algorithm we tested using the function (3), and its effectiveness is evaluated by adjusting ε. MI-FGSM is an upgraded version of FGSM that used in this paper, the number of iterations T and momentum degree μ is two other hyperparameters to be controlled instead of ε. We then made another evaluation with Deepfool and tested our own modified method based on MI-FGSM. The Deepfool automatically computes the minimum perturbations without setting up a fixed ε. Since it has been shown that iterative methods are stronger white-box adversaries than one-step methods at the cost of worse transferability, our method can keep the transferability to a certain extent.

3.2.1 Deepfool

In our case, fingerprint liveness detection is always treated as a binary classification problem, and therefore the Deepfool algorithm is used here for binary classifiers as well. The author assumes $\hat{k}(\boldsymbol{x}) = \text{sign}(f(\boldsymbol{x}))$ where f represents a binary image classification function and derives the general algorithm, which can be applied to any differentiable binary classifier f. That is, to adopt an iterative process to estimate $\Delta(\boldsymbol{x}; f)$. Specifically, f is linearized around the current point $\boldsymbol{x}_i$ at each iteration where i is the current number of iterations, and the minimal perturbation of linearized f can be computed through:

$$\text{argmin } \|r_i\|_2 \text{ subject to } f(\boldsymbol{x}_i) + \nabla f(\boldsymbol{x}_i)^T r_i = 0 \qquad (4)$$

The algorithm terminates when $\boldsymbol{x}_i$ changes sign of the classifier's result or maximum iterations is reached. The Deepfool algorithm for binary classifiers is summarized as follows.

Algorithm Deepfool

Input: Image $\boldsymbol{x}$, classifier f
Output: Perturbation $\hat{\boldsymbol{r}}$

Initialize $\boldsymbol{x}_0 \leftarrow \boldsymbol{x}$, $i \leftarrow 0$
While $\text{sign}((f(\boldsymbol{x}_i)) = sign((f(\boldsymbol{x}_0))$ **do**
  $\boldsymbol{r}_i \leftarrow -\frac{f(\boldsymbol{x}_i)}{\|\nabla f(\boldsymbol{x}_i)\|_2^2}\nabla f(\boldsymbol{x}_i)$
  $\boldsymbol{x}_{i+1} \leftarrow \boldsymbol{x}_i + \boldsymbol{r}_i$
  $i \leftarrow i + 1$
End while
Return $\hat{\boldsymbol{r}} = \Sigma_i \boldsymbol{r}_i$

3.2.2 Momentum iterative fast gradient sign method

Momentum iterative fast gradient sign method (MI-FGSM) is upgraded twice in the basic version of FGSM. The I-FGSM iteratively applies multiple steps with a small step size α, and MI-FGSM further introduces momentum [41]. Momentum method is a technique to accelerate and stabilize stochastic gradient descent algorithm. Gradients in the previous iteration are accumulated in the current gradient direction of the loss function, it can be considered a velocity vector pass through every iteration. Dong et al. first applied the technique of momentum to generate adversarial samples and get tremendous benefits. The MI-FGSM is summarized below.

Algorithm MI-FGSM

Input1: A classifier f with loss J; a clean image $\boldsymbol{x}$ and ground-truth y_{true};
Input2: The size of perturbation ε; iterations T and decay factor μ.
Output: An adversarial sample x^* with $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty \le \varepsilon$

$\alpha = \frac{\varepsilon}{T}$;
$\boldsymbol{g}_0 = 0$; $\boldsymbol{x}_0^* = \boldsymbol{x}$;
For $t = 0$ to $T - 1$ **do**
  Input $\boldsymbol{x}_t^*$ to f and obtain the gradient $\nabla J(\boldsymbol{x}_t^*, y_{true})$
  Update $\boldsymbol{g}_{t+1}$ by accumulating the velocity vector in the gradient direction as
  $$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_x J(\boldsymbol{x}_t^*, y_{true})}{\|\nabla_x J(\boldsymbol{x}_t^*, y_{true})\|_1};$$
  Update $\boldsymbol{x}_{t+1}^*$ by applying the sign gradient as
  $$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_t^* + \alpha \cdot sign(\boldsymbol{g}_{t+1});$$
End for
Return $\boldsymbol{x}^* = \boldsymbol{x}_T^*$

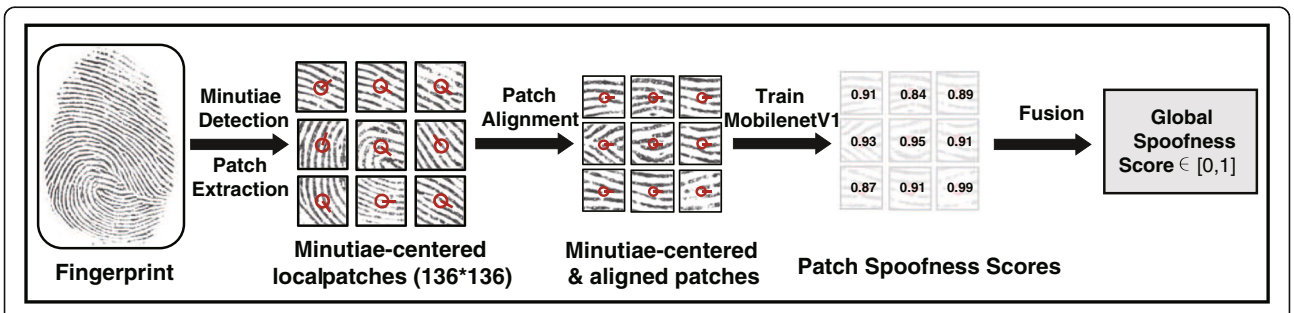

Fig. 3 The flow chart of Chugh's method

3.2.3 Transformation robust attack

During the experiments, we found that adversarial samples generated by these methods are not robust enough to image transformations, for instance, resize, horizontal flip, and rotations. However, these transformations commonly occur in the physical world, and to generate adversarial samples that can successfully attack detection modules under any conditions, we have to take such demand into account. A heuristic and natural idea is to add slight Gaussian noise in order to disturb the sample at every iteration. And by randomly rotating the sample at a very small angle, we could improve its robustness to rotation transformations and even transferability on a different model. Note that with the addition of the noise, the global perturbation degree is increased compared to the original MI-FGSM.

Algorithm Transformation Robust Attack(TRA)

Input1: A classifier f with loss J; a clean image $\boldsymbol{x}$ and ground-truth y_{true};
Input2: The size of perturbation ε; iterations T and decay factor μ.
Output: An adversarial example x^* with $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty$

$\alpha = \frac{\varepsilon}{T}$;
$\boldsymbol{g}_0 = 0$; $\boldsymbol{x}_0^* = \boldsymbol{x}$;
For $t = 0$ to $T - 1$ **do**
  Input $\boldsymbol{x}_t^*$ to f and obtain the gradient $\nabla J(\boldsymbol{x}_t^*, y_{true})$
  Update $\boldsymbol{g}_{t+1}$ by accumulating the velocity vector in the gradient direction as
  $$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \nabla_x J(\boldsymbol{x}_t^*, y_{true})$$
  Update $\boldsymbol{x}_{t+1}^*$ by applying the sign gradient as
  $$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_t^* + \alpha \cdot \mathrm{sign}(\boldsymbol{g}_{t+1})$$
  Adding gaussian noise to $\boldsymbol{x}_{t+1}^*$
  $$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_{t+1}^* + X;\ X \sim N(0,\ 0.01)$$
  Random rotation $\boldsymbol{x}_{t+1}^*$ by randomly (-5,5) degree
End for
Return $\boldsymbol{x}^* = \boldsymbol{x}_T^*$

4 Results and discussion

In this section, we conduct different adversarial attacks on the above models. Details are available in the following part. In general, we compare the success rates of different attack methods to different models, and furthermore, evaluate their robustness to varies transformations such as rotating and resizing.

4.1 Datasets

The fingerprint datasets used in this paper are from Liveness Detection Competition (LivDet), containing the years 2013 [42] and 2015 [43], namely, LiveDet2013 and LiveDet2015(Table 1). The earlier competition datasets are not used because of fingerprint images quality and the coincidence of data distribution, e.g., fake fingerprints made with the same materials captured by the same sensors probably cause similar results. LivDet 2013 consists of fingerprint images captured by four different sensors. Each has approximately 2000 images of fake/real fingerprints respectively, the number of real/fake fingerprints ratio is also equally distributed between training and testing sets. The fake fingerprints are made from different materials: Gelatin, Latex, Eco Flex, Wood Glue, and Modasil. Although the sizes of the images range from 315 × 372 to 700 × 850 pixels depending on sensors, they were all resized concerning the input dimension of the models which is 224 × 224 for VGG and 227 × 227 pixels for Alexnet.

4.2 Settings

We adjust ε in FGSM to control the perturbation degree, five different values: 0.03, 0.06, 0.09, 0.12, and 0.15 are tested on all three detection algorithms trained on LivDet2013 in a white-box manner. Since Deepfool automatically searches the minimum perturbation, it does not restrict the perturbation degree, however, we limit the number of max iterations as 100 to guarantee time consumption acceptable, also, 100 is a moderate value that ensures most fingerprint images can be converted to their adversarial samples. As for MI-FGSM, we set the $\varepsilon = 0.12$, iterations = 10, and decay factor = 0.5 according to the existing literature and our preliminary test results. Our method originates in MI-FGSM, thus we apply similar settings but iterations number raised to 20. The noise added obeys gauss distribution of which the standard deviation is 0.1 and the mean is 0. Meanwhile, we set the angle of random rotation between – 5° and 5°.

To evaluate the feasibility of black-box attack, we have trained our own detection models. We first consider two models of which one is shallow with several layers and the other is much deeper. The shallow one consists of 4 convolutional layers with 3×3 kernel, and the stride is 2 thus no pooling layer is involved. Each layer is twice as deep as the previous one while there are 32 channels in first layer. The deeper one consists of 5 blocks in which there are 3 convolutional layers and BN layers, numbers of kernels in block are doubled to previous layer and consistent inside the block as it is 32 in the first one. We further train two ensemble models with three branches

Table 1 Summary of liveness detection datasets used in our work

Dataset	LivDet 2013			LivDet 2015		
Fingerprint reader	Biometrika	ItalData	Crossmatch	GreenBit	Biometrika	CrossMatch
Image size	315 × 372	640 × 480	800 × 750	500 × 500	1000 × 1000	640 × 480
DPI	569	500	500	500	1000	500
Live images train/test	1000/1000	1000/1000	1250/1250	1000/1000	1000/1000	1510/1500
Spoof images train/test	1000/1000	1000/1000	1000/1000	1000/1500	1000/1500	1473/1448
Cooperative	No	No	Yes	Yes	Yes	Yes

Table 2 Success rate of FGSM attacks with different ε in white-box manner. Bio2013, Ita2013, and Cro2013 represent dataset of Biometrika, ItalData, and Crossmatch in LiveDet2013, respectively

Model and dataset	Perturbation degree				
	ε = 0.03 (%)	ε = 0.06 (%)	ε = 0.09 (%)	ε = 0.12 (%)	ε = 0.16 (%)
VGG19 (Bio2013)	54.5	64.9	75.6	84.4	98.5
Alexnet(Bio2013)	61.5	69.6	76.0	86.6	99.8
MobilenetV1(Bio2013)	48.4	61.1	72.2	85.4	97.3
VGG19 (Ita2013)	51.8	58.9	64.3	80.3	96.1
Alexnet(Ita2013)	56.7	58.4	74.7	81.7	98.4
MobilenetV1 (Ita2013)	41.3	56.0	65.2	75.4	97.9
VGG19 (Cro2013)	54.3	61.5	68.2	85.1	96.6
Alexnet(Cro2013)	58.6	62.4	69.6	75.6	93.7
MobilenetV1(Cro2013)	45.7	55.2	74.8	78.6	89.3

for each in addition to the models used above: one is shallow and the other is relatively deep as well. Each branch is different to each other regarding size of kernel, number of kernels and pooling methods. This idea is originated in inception module. The reason we set up the black-box attack models to be ensemble is that successful attacks on a collection of models may cause the improvement on attacking single model. This is a natural intuition and has been verified in our work. Specific structures of the above four models are different for different dataset and chosen via an extensive search. At last, we prepared five kinds of transformations to research their influence. Resizing represents that we expand the adversarial sample by 2X and restore it to original size, this approximately equal to adding very small noise according to scaling method. We also horizontally flip and rotate the samples at random angle −30° to 30°, combination of resize and flip and resize and rotation are also considered.

Table 3 Success rate of different methods in white-box manner. Gre2015, Bio2015, and Cro2015 represent dataset of GreenBit, Biometrika, and CrossMatch in LiveDet2015 respectively

Model and dataset	Method			
	FGSM (%)	Deepfool (%)	MI-FGSM (%)	TRA (%)
VGG19 (Gre2015)	87.4	97.5	99.0	98.2
Alexnet(Gre2015)	85.2	97.9	99.6	99.1
MobilenetV1(Gre2015)	80.3	98.5	97.2	98.2
VGG19 (Bio2015)	72.6	95.3	98.2	96.3
Alexnet(Bio2015)	82.7	97.1	97.4	98.0
MobilenetV1(Bio2015)	79.0	98.9	96.5	97.2
VGG19 (Cro2015)	87.4	96.4	98.0	96.0
Alexnet(Cro2015)	76.6	97.8	97.3	98.1
MobilenetV1(Cro2015)	80.1	97.5	97.1	97.9

4.3 Results

We first evaluate original FGSM and results are shown in Table 2, this one-step attack methed does not produce a satisfactory effect on target models in white-box manner with a low perturbation degree. The table shows that while ε = 0.03, almost over half inputs can be turned into adversarial samples which lead to misclassifications. With the ε increasing, the ratio unsurprisingly increases and is nearly full at 0.16. However, with larger ε, the attack success rate obviously rises, we did not further improve the value of ε because it is foreseeable that 100% is reachable with a ε large enough. We deem that this increase in the success rate is at the expense of larger perturbations. We also find some other notable phenomena from the table. Generally, under the same ε, models with greater complexity are more robust to adversarial attacks even in white-box, the complexity here is depth. It may be due to the high dimension of complex model and the learned decision boundary is complex as well. Another reasonable explanation is that as the complexity of the model increases, its learning ability becomes stronger, therefore its adversarial samples are harder to make. We also found that fingerprint images of higher resolution are always harder to be made into

Table 4 Average robustness computed for different methods. For each model, we randomly pick 200 samples from GreenBit, Biometrika, and CrossMatch in LivDet2015, respectively, and compute their average robustness

Methods	Models		
	VGG19	Alexnet	Mobilenet
Deepfool	3.9×10^{-2}	3.4×10^{-2}	4.7×10^{-2}
FGSM(ε = 0.12)	8.7×10^{-2}	7.4×10^{-2}	9.1×10^{-2}
MI-FGSM	4.0×10^{-2}	3.7×10^{-2}	4.4×10^{-2}
TRA	4.3×10^{-2}	4.2×10^{-2}	4.6×10^{-2}

Table 5 Error rate of different models on Biometrika2013 and Biometrika2015

Datasets	Model			
	ShallowCNN (%)	DeepCNN (%)	EnsembleCNN(shollow) (%)	EnsembleCNN(deep) (%)
Bio2013	6.7	6.5	4.3	4.3
Bio2015	5.9	4.4	4.1	3.7

adversarial samples, as high resolution provides more discriminative details.

An overall evaluation of different attack methods in white-box manner shows their performance in Table 3. Here we set $\varepsilon = 0.12$ of FGSM, and other settings are the same as mentioned above. It shows that the iterative method is generally much better than FGSM, although the MI-FGSM and our method both set $\varepsilon = 0.12$ as well. It can be also observed that the attack success rate of the high-resolution dataset is slightly lower than that of the lower resolution dataset too. In a white-box manner, our method achieves competitive results compared to other iterative algorithms.

To research the average perturbation degree of the adversarial samples generated by different methods, we compute the "average robustness" using the method proposed in [27]. It is defined by

$$\frac{1}{|N|}\sum_{\boldsymbol{x}\in N}\frac{\|\hat{\boldsymbol{r}}(\boldsymbol{x})\|_2}{\|\boldsymbol{x}\|_2} \tag{5}$$

where $\hat{\boldsymbol{r}}(\boldsymbol{x})$ is the perturbation computed by different methods, and N denotes the dataset. This computes the average perturbation amplitude by averaging the proportion of perturbation vector to the original image of each adversarial sample. We report in Table 4 the average robustness of each model and method. FGSM requires the largest perturbations to successfully generate an adversarial sample. Our method gets similar results to Deepfool and MI-FGSM, a much lower average perturbation degree. It is consistent with our previous observation that a deeper and complicated network is more robust to adversarial samples, and more perturbations are necessary for a successful attack. It also shows that the magnitude of the disturbance caused by our method is acceptable and at the same level compared to other advanced methods. Moreover, the TRA is not seriously affected by the complexity of the target model and the average robustness is stable among different target models.

All the above experiments are white-box attacks; we conduct more experiments under black-box condition. We first trained four models which have different structures to each other and to the target models. Table 5 shows their performance of detecting fake fingerprints on Biometrika2013 and Biometrika2015. And in Table 6, we report the attack success rate of black-box with adversarial samples generated from our models. It is tested on Biometrika2013 and Biometrika2015 to further analyze the influence of image resolution on attack success rate. Attack success rate of black-box is much lower than that of white-box; however, with the depth increasing, the success rate improves. Compared to single CNN, the ensemble models no matter shallow or deep both achieve considerable performances, adversarial samples generated by them are more likely to realize black-box attacks. The influence of complexity of target models on attack success rate is more significant in this case, MobilenetV1 is the hardest one to attack while Alexnet is easier. This part of the experiments also

Table 6 Black-box attacks with adversarial samples generated from different models by MI-FGSM and TRA

Generation model	Target model&Datasets					
	VGG19 Bio2013 (%)	Alexnet Bio2013 (%)	MobilenetV1 Bio2013 (%)	VGG19 Bio2015 (%)	Alexnet Bio2015 (%)	MobilenetV1 Bio2015 (%)
	MI-FGSM					
ShollowCNN	5.6	7.5	4.2	4.7	6.9	3.7
DeepCNN	8.9	10.4	6.5	7.1	11.2	5.4
ShallowEnsemble	21.3	26.1	7.3	19.4	27.3	5.2
DeepEnsemble	21.9	27.0	8.9	22.4	25.1	7.6
	TRA					
ShollowCNN	6.6	8.3	5.2	5.6	6.6	3.4
DeepCNN	9.0	12.7	6.4	8.1	12.3	5.6
ShallowEnsemble	23.3	27.9	8.5	22.1	27.4	6.7
DeepEnsemble	20.5	27.1	10.2	21.4	24.2	7.7

Table 7 Robustness to transformations of different adversarial attack methods, we randomly pick 300 samples from GreenBit, Biometrika, and CrossMatch in LivDet2015, respectively, and generate their corresponding adversarial samples to attack VGG19

Transformation	Methods and models					
	MI-FGSM			TRA		
	(White-box) (%)	(DeepCNN) (%)	(Deepensemble) (%)	(White-box) (%)	(DeepCNN) (%)	(Deepensemble) (%)
Original	99.0	7.9	22.0	98.2	11.0	24.1
Resize	84.5	5.3	6.6	87.6	7.1	9.0
Flip	92.1	5.4	5.8	93.0	6.4	6.8
Rotation	47.5	1.1	2.3	56.1	4.5	4.4
Resize and flip	78.3	0.2	3.4	84.2	1.7	5.9
Resize and rotation	33.2	0.2	0.9	33.7	0.9	4.1

proves that fingerprint images of higher resolution provide more discriminative cues for models to learn better features and lead to more robust of the models to adversarial samples.

For a more comprehensive assessment in the feasibility that attack deep learning-based fingerprint liveness detection algorithms deployed in the physical world. We also compared our method and MI-FGSM in both white-box and black-box manners while various transformations applied in the adversarial samples. Table 7 shows that even under white-box, there is still a great probability to make adversarial samples invalid. And about half adversarial samples will be classified correctly after rotations, and most of them are invalid after resizing and rotations. These transformations are more destructive in black-box attacks; however, a small part of adversarial samples generated by our method can survive. Our method surpasses MI-FGSM by a narrow margin in various situations. It indicates that these detection algorithms still may be threatened in complex cases like this.

5 Conclusion

In this work, we provided extensive experimental evidence that cheating excellent deep learning-based fingerprint liveness detection schemes by adversarial samples is feasible. These detection networks could be easily break through by basic FGSM in white-box manner at the cost of some perturbations. With more advanced methods like Deepfool and MI-FGSM, almost arbitrary fingerprint image can turn into an adversarial sample with more imperceptible changes. We note that adversarial samples generated by the above methods are not robust enough to transformations, for instance, resize, horizontal flip, and rotations. Thus, we also proposed an algorithm to generate adversarial samples that are slightly more robust to various transformations by adding noise and random rotations during every iteration. These methods are evaluated on LivDet2013 and LivDet2015 datasets. According to our results, a small part of adversarial samples possesses transferability on different models, that indicate it is also possible to cause misclassification under black-box scenarios. In terms of robustness to transformations, further evaluations demonstrate the proposed method can also surpass others slightly. These results highlight the potential risks of existing fingerprint liveness detection algorithms, and we hope our work will encourage researchers to start designing more robust detection algorithms that have innate adversarial robustness to achieve higher security.

Abbreviations

X_c: An original clean image; x^*: An adversarial sample; y_{true}: The label of the original clean image; y_{target}: The target label; p: Perturbation; f: The classifier; $f(x_c)$: Classification result; $J(\ldots)$: Loss function of the classifier; θ: Parameter of the classifier; ε: The size of perturbation ε; T: Iterations; μ: Decay factor

Acknowledgements

This work is supported in part by the Jiangsu Basic Research Programs-Natural Science Foundation under grant numbers BK20181407, in part by the National Natural Science Foundation of China under grant numbers 61672294, in part by Six peak talent project of Jiangsu Province (R2016L13), Qing Lan Project of Jiangsu Province and "333" project of Jiangsu Province, in part by the National Natural Science Foundation of China under grant numbers U1836208, 61502242, 61702276, U1536206, 61772283, 61602253, 61601236, and 61572258, in part by National Key R&D Program of China under grant 2018YFB1003205, in part by NRF-2016R1D1A1B03933294, in part by the Jiangsu Basic Research Programs-Natural Science Foundation under grant numbers BK20150925 and BK20151530, in part by Humanity and Social Science Youth Foundation of Ministry of Education of China (15YJC870021), in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund, in part by the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET) fund, China. Zhihua Xia is supported by BK21+ program from the Ministry of Education of Korea.

Authors' contributions

JF and ZX collectively designed the research, performed the research and wrote the paper.
PY partly performed the research, analyzed the data, and partly wrote the paper.
FX partly designed the research, wrote the paper, and modified the paper.
All authors read and approved the final manuscript.

Authors' information

Jianwei Fei received his BE degree in Electronic and Information Engineering from Nanjing Forestry University in 2014. He is currently pursuing a master's degree in Computer Science in Nanjing University of Information Science and Technology. His reach interests include artificial intelligence security and multimedia forensics.

Zhihua Xia received a BS degree in Hunan City University, China and PhD degree in computer science and technology from Hunan University, China, in 2006 and 2011, respectively. He works as an associate professor in the School of Computer and Software, Nanjing University of Information Science and Technology. His research interests include digital forensic and encrypted image processing. He is a member of the IEEE from 1 March 2014.
Peipeng Yu is a BE (2019) and is currently pursuing master degree in Computer Science in Nanjing University of Information Science and Technology. His reach interests include artificial intelligence security.
Fengjun Xiao received his BS degree in Economics from the BeiHang University in 2009. He received his Master's degree in Technology Policy in 2014 under the supervision of Prof. Shi Li. He has been a Doctoral Candidate under the supervision of Prof. Chengzhi Li and began his research on the Network Security and Emergency Management since 2015.

Funding
This work is funded by the National Natural Science Foundation of China under grant numbers 61672294.

Availability of data and materials
The datasets used and analyzed during the current study are available from the first author on reasonable request.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Jiangsu Engineering Center of Network Monitoring, Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China. [2]Hangzhou Dianzi University, No. 1, 2nd Street Jianggan District, Hangzhou City, Zhejiang Province, China.

Received: 31 October 2019 Accepted: 31 December 2019
Published online: 13 January 2020

References

1. Y. Zheng, X. Xu, L. Qi, Deep CNN-assisted personalized recommendation over big data for mobile wireless networks. Wireless Communications and Mobile Computing **2019** (2019)
2. Y. Zheng, J. Zhu, W. Fang, L.-H. Chi, Deep learning hash for wireless multimedia image content security. Security and Communication Networks **2018** (2018)
3. H. Wang et al., in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Cosface: large margin cosine loss for deep face recognition (2018), pp. 5265–5274
4. K. Cao, Y. Rong, C. Li, X. Tang, C. Change Loy, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Pose-robust face recognition via deep residual equivariant mapping (2018), pp. 5187–5196
5. Y. Sun, D. Liang, X. Wang, X. Tang, Deepid3: Face recognition with very deep neural networks. arXiv preprint arXiv **1502**, 00873 (2015)
6. Z. Zheng, X. Yang, Z. Yu, L. Zheng, Y. Yang, J. Kautz, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Joint discriminative and generative learning for person re-identification (2019), pp. 2138–2147
7. Y. Li, C. Huang, C.C. Loy, X. Tang, in *European Conference on Computer Vision*. Human attribute recognition by deep hierarchical contexts (Springer, 2016), pp. 684–700
8. P. Li, X. Chen, S. Shen, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Stereo r-cnn based 3d object detection for autonomous driving (2019), pp. 7644–7652
9. F. Codevilla, M. Miiller, A. López, V. Koltun, A. Dosovitskiy, in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. End-to-end driving via conditional imitation learning (IEEE, 2018), pp. 1–9
10. C. Szegedy et al., Intriguing properties of neural networks. arXiv preprint arXiv **1312**, 6199 (2013)
11. A. Krizhevsky, I. Sutskever, G.E. Hinton, in *Advances in neural information processing systems*. Imagenet classification with deep convolutional neural networks (2012), pp. 1097–1105
12. Z. Xia, L. Jiang, D. Liu, L. Lu, B. Jeon, BOEW: a content-based image retrieval scheme using bag-of-encrypted-words in cloud computing. IEEE Transactions on Services Computing (2019)
13. Z. Xia, L. Lu, T. Qiu, H. Shim, X. Chen, B. Jeon, A privacy-preserving image retrieval based on AC-coefficients and color histograms in cloud environment. Computers, Materials & Continua **58**(1), 27–44 (2019)
14. Z. Xia, L. Jiang, X. Ma, W. Yang, P. Ji, N. Xiong, A privacy-preserving outsourcing scheme for image local binary pattern in secure industrial internet of things. IEEE Transactions on Industrial Informatics (2019)
15. Z. Xia, N.N. Xiong, A.V. Vasilakos, X. Sun, EPCBIR: an efficient and privacy-preserving content-based image retrieval scheme in cloud computing. Information Sciences **387**, 195–204 (2017)
16. Z. Xia, Y. Zhu, X. Sun, Z. Qin, K. Ren, Towards privacy-preserving content-based image retrieval in cloud computing. IEEE Transactions on Cloud Computing **6**(1), 276–286 (2015)
17. M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition, in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security 2016, pp. 1528-1540: ACM.
18. M. Sharif, S. Bhagavatula, L. Bauer, M.K. Reiter, Adversarial generative nets: Neural network attacks on state-of-the-art face recognition. arXiv preprint arXiv **1801**, 00349 (2017)
19. G. Goswami, N. Ratha, A. Agarwal, R. Singh, M. Vatsa, in *Thirty-Second AAAI Conference on Artificial Intelligence*. Unravelling robustness of deep learning based face recognition against adversarial attacks (2018)
20. H. Tang, X. Qin, *Practical methods of optimization* (Dalian University of Technology Press, Dalian, 2004), pp. 138–149
21. A. Kurakin et al., in *The NIPS'17 Competition: Building Intelligent Systems*. Adversarial attacks and defences competition (Springer, 2018), pp. 195–231
22. W. Brendel et al., Adversarial vision challenge. arXiv preprint arXiv **1808**, 01976 (2018)
23. I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples. arXiv preprint arXiv **1412**, 6572 (2014)
24. T. Miyato, S.-i. Maeda, M. Koyama, S. Ishii, Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE transactions on pattern analysis and machine intelligence **41**(8), 1979–1993 (2018)
25. A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world. arXiv preprint arXiv **1607**, 02533 (2016)
26. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Rethinking the inception architecture for computer vision (2016), pp. 2818–2826
27. S.-M. Moosavi-Dezfooli, A. Fawzi, P. Frossard, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Deepfool: a simple and accurate method to fool deep neural networks (2016), pp. 2574–2582
28. Y. Dong et al., in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Boosting adversarial attacks with momentum (2018), pp. 9185–9193
29. J. Su, D.V. Vargas, K. Sakurai, One pixel attack for fooling deep neural networks. IEEE Transactions on Evolutionary Computation (2019)
30. K. Eykholt *et al.*, Robust physical-world attacks on deep learning models, 2018.
31. A.J. Bose, P. Aarabi, *Adversarial attacks on face detectors using neural net based constrained optimization* (2018)
32. Z. Xia, C. Yuan, R. Lv, X. Sun, N.N. Xiong, Y.-Q. Shi, *A novel weber local binary descriptor for fingerprint liveness detection, IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2018)
33. R.F. Nogueira, R. de Alencar Lotufo, R.C. Machado, in *2014 IEEE workshop on biometric measurements and systems for security and medical applications (BIOMS) Proceedings*. Evaluating software-based fingerprint liveness detection using convolutional networks and local binary patterns (IEEE, 2014), pp. 22–29
34. R.F. Nogueira, R. de Alencar Lotufo, R.C. Machado, Fingerprint liveness detection using convolutional neural networks. IEEE transactions on information forensics and security **11**(6), 1206–1213 (2016)
35. T. Chugh, K. Cao, A.K. Jain, Fingerprint spoof buster: Use of minutiae-centered patches. IEEE Transactions on Information Forensics and Security **13**(9), 2190–2202 (2018)

36. S. Kim, B. Park, B.S. Song, S. Yang, Deep belief network based statistical feature learning for fingerprint liveness detection ☆. Pattern Recognition Letters **77**(C), 58–65 (2016)
37. T. Nguyen, E. Park, X. Cui, V. Nguyen, H. Kim, fPADnet: small and efficient convolutional neural network for presentation attack detection. Sensors **18**(8), 2532 (2018)
38. F. Pala, B. Bhanu, in *Deep Learning for Biometrics*. Deep triplet embedding representations for liveness detection (Springer, 2017), pp. 287–307
39. I. Sutskever, J. Martens, G. Dahl, G. Hinton, in *International conference on machine learning*. On the importance of initialization and momentum in deep learning (2013), pp. 1139–1147
40. C. Kai, E. Liul, L. Pangi, J. Liangi, T. Jie, in *International Joint Conference on Biometrics*. Fingerprint matching by incorporating minutiae discriminability (2011)
41. B.T. Polyak, Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics **4**(5), 1–17 (1964)
42. L. Ghiani et al., in *Iapr International Conference on Biometrics*. LivDet 2013 Fingerprint Liveness Detection Competition 2013 (2013)
43. L. Ghiani, D.A. Yambay, V. Mura, G.L. Marcialis, F. Roli, S.A. Schuckers, Review of the Fingerprint Liveness Detection (LivDet) competition series: 2009 to 2015. *Image and Vision Computing* **58**, 110–128 (2017)

Publisher's Note