Electronic Research Archive

http://www.aimspress.com/journal/era

ERA, 32(10): 5592–5614.
DOI: 10.3934/era.2024259
Received: 20 May 2024
Revised: 31 August 2024
Accepted: 09 September 2024
Published: 11 October 2024

Research article

Learning deep forest for face anti-spoofing: An alternative to the neural network against adversarial attacks

Rizhao Cai[1], Liepiao Zhang[2,4], Changsheng Chen[3,*], Yongjian Hu[4] and Alex Kot[1]

[1] School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore
[2] GRGTally-vision I.T. Co., Ltd., Guangzhou 510663, China
[3] College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518061, China
[4] School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510641, China

* **Correspondence:** Email: cschen@szu.edu.cn.

Abstract: Face anti-spoofing (FAS) is significant for the security of face recognition systems. neural networks (NNs), including convolutional neural network (CNN) and vision transformer (ViT), have been dominating the field of the FAS. However, NN-based methods are vulnerable to adversarial attacks. Attackers could insert adversarial noise into spoofing examples to circumvent an NN-based face-liveness detector. Our experiments show that the CNN or ViT models could have at least an 8% equal error rate (EER) increment when encountering adversarial examples. Thus, developing methods other than NNs is worth exploring to improve security at the system level. In this paper, we have proposed a novel solution for FAS against adversarial attacks, leveraging a deep forest model. Our approach introduces a multi-scale texture representation based on local binary patterns (LBP) as the model input, replacing the grained-scanning mechanism (GSM) used in the traditional deep forest model. Unlike GSM, which scans raw pixels and lacks discriminative power, our LBP-based scheme is specifically designed to capture texture features relevant to spoofing detection. Additionally, transforming the input from the RGB space to the LBP space enhances robustness against adversarial noise. Our method achieved competitive results. When testing with adversarial examples, the increment of EER was less than 3%, more robust than CNN and ViT. On the benchmark database IDIAP REPLAY-ATTACK, a 0% EER was achieved. This work provides a competitive option in a fusing scheme for improving system-level security and offers important ideas to those who want to explore methods besides CNNs. To the best of our knowledge, this is the first attempt at exploiting the deep forest model in the problem of FAS, with the consideration of adversarial attacks.

Keywords: deep forest; adversarial attacks; face anti-spoofing

1. Introduction

Face recognition systems, which identify an individual with her/his face, have been widely used in practical applications such as mobile phone unlocking. However, the existing face recognition techniques cannot differentiate between genuine faces (captured from humans) and spoofing faces (captured from the faces in images, digital displays, etc.). Most of the face recognition systems are therefore vulnerable to Presentation Attack (PA), including print attack, and replay attack. Attackers could bypass the face recognition systems by presenting different types of spoofing faces since face images can be readily available to attackers from social platforms, e.g., Facebook, Instagram [1]. To guarantee the security of face recognition systems, there are increasing demands for developing face anti-spoofing (FAS) techniques.

Traditionally, image descriptors, such as the local binary pattern (LBP) and scale invariant feature transform (SIFT), are utilized to extract features for describing the data from the FAS databases. Recently, with the powerful ability for learning deep representations from data, convolutional neural networks (CNNs) have been successfully exploited in various visual tasks, e.g., objects classification [2], face recognition [3], etc., and have achieved the state-of-the-art performances. As an excellent competitor, vision transformers (ViT) [4] with the unique self-attention mechanism, have surpassed CNNs in certain tasks, such as image recognition [5]. CNNs have been successfully applied in FAS, with significant improvement in architectures and optimization [6–9]. Also, advanced methods-based vision transformers for FAS have also been proposed with specific adapters [10–12].

Both CNN and ViT methods are based on neural networks, and they have shown excellent capacities in learning representative features. However, the NN methods are vulnerable to adversarial attack [13, 14]. Under such an adversarial attack, NN models would fail to correctly classify the adversarial examples, which are generated by imposing some human-invisible perturbations on the original samples. What is more, though adversarial examples are usually manipulated in the digital world, they could still take effect even after a print-and-capture cycle [15–17]. In other words, the adversarial attack can be conducted in the physical world. Worse still, the adversarial examples are shown to be transferable [18–21]. Adversarial examples can be transferred to attack other models as long as they adopt the same or similar features even if the classification models are different (support vector machine, random forest, etc.) [21]. Therefore, it is likely for attackers to generate adversarial-spoofing examples to attack an NN model, either CNN or ViT for face-liveness detection in a face recognition system.

Fortunately, using handcrafted feature-based methods could be a solution. In [20, 21], it is revealed that adversarial examples are non-transferable when they are in different feature spaces as the input of their victim models. This indicates that the handcrafted features from RGB images as input for a face anti-spoofing model could be an approach against the adversarial-spoofing attack targeted at the CNN-based models. In cybersecurity applications, it is also suggested in [22] that ensembling a diverse pool of models of different features could improve the security of a cyber system against adversarial attacks. Hence, to alleviate the threats of adversarial attacks, handcrafted feature-based methods also deserve efforts of exploration. However, previous works using support vector machine (SVM) as classifiers are not discriminative enough. Therefore, exploring classifiers other than neural networks to fight against adversarial attacks is important.

In this paper, we introduce a new feature-based method, deep forest [23], for the FAS problem. Deep

forest is an advanced tree-ensemble method. It consists of the grained-scanning mechanism (GSM) for learning representations from data, which scans pixels with slide windows. The scanned pixels are forwarded into an ensemble of random forests to get the outputs. These outputs are concatenated and will go through a cascade of several layers of forest models. Deep forest has been evaluated on several visual tasks, e.g., face recognition, handwriting recognition, etc., and it achieves competitive performance [23]. However, we found that the raw pixel scanned by the vanilla GSM used in [23] is not effective in representing spoofing features for the face anti-spoofing problem. Since spoofing features can be represented by texture features [24–26], we re-devise the representation construction by employing an LBP to construct the multi-scale texture representations. Experimental results show that the proposed approach has achieved competitive performance.

• To the best of our knowledge, this is the first work that introduces deep forest to the problem of FAS. Our method offers an important reference and a competitive option to those who want to fuse diverse methods in their schemes for system-level security in their cases.

• We re-devise the representation construction by utilizing the LBP descriptors instead of the GSM. The proposed scheme that integrates LBP descriptors and the deep forest learning method achieves better results than that of the GSM [23].

• The proposed scheme shows competitive performance compared to state-of-the-art approaches. On the IDIAP REPLAY-ATTACK database [27], a 0% equal error rate (EER) is achieved. Also, extensive experiments on the two benchmark databases, the MSU USSA database and the ROSE-YOUTU database, have been conducted. On the MSU database, an EER of 1.56% is obtained, which is a competitive result compared to the patch-based CNN (0.55% EER) and the depth-based CNN (2.62% EER) proposed by [28].

The rest of the paper is organized as follows: Section 2 presents brief literature reviews about approaches to FAS and about learning methods that are forest-related. The proposed scheme is elaborated in Section 3. The performance of the proposed scheme is evaluated in Section 4. Finally, Section 5 concludes this paper.

2. Related works

In this section, the literature on both traditional handcrafted feature-based methods and CNN-based methods for the problem of FAS is first reviewed, followed by the tree-ensemble learning methods.

2.1. The existing works on FAS

2.1.1. The traditional methods

Most of the traditional FAS approaches focus on designing handcrafted features and learning classifiers with traditional learning methods, e.g., SVM. Texture analysis is one of the main approaches to spoofing face detection since there are inherent texture disparities between genuine faces and spoofing faces of the print attack or of the replay attack. As can be seen in Figure 1, images of the spoofing faces, compared to the genuine faces, usually have lower quality and contain visual artifacts because of the recapturing process. These disparities can be described effectively by texture descriptors. Relevant methods aimed at capturing these disparities in the Fourier spectrum or spatial domain are reported. Reference [29] uses difference-of-gaussian (DoG) features to describe the

disturbance of frequency resulting from the recapturing. Besides, the local phase quantization (LPQ) that analyzes distortion through the phase is also discussed by [30]. In addition, in the spatial domain, a significant number of research works employ the LBP-based features to describe the disparities from local texture information [24–26]. Analogously, methods that utilize the scale-invariant feature transform (SIFT) and speed-up robust feature [31] are also reported. Besides, to utilize motion information from the temporal domain, the texture-based methods mentioned above are extended into three orthogonal planes, e.g., LBP-TOP [32] and LPQ-TOP [33]. Moreover, the color information of spoofing faces, which is less abundant after distortions in the recapturing process, is essential in discriminating spoofing faces. Therefore, color texture methods are proposed in [34] by extracting features from separate channels in a certain color space (e.g., to extract features of images in HSV space from the three components H, S, and V individually) using the aforementioned methods.

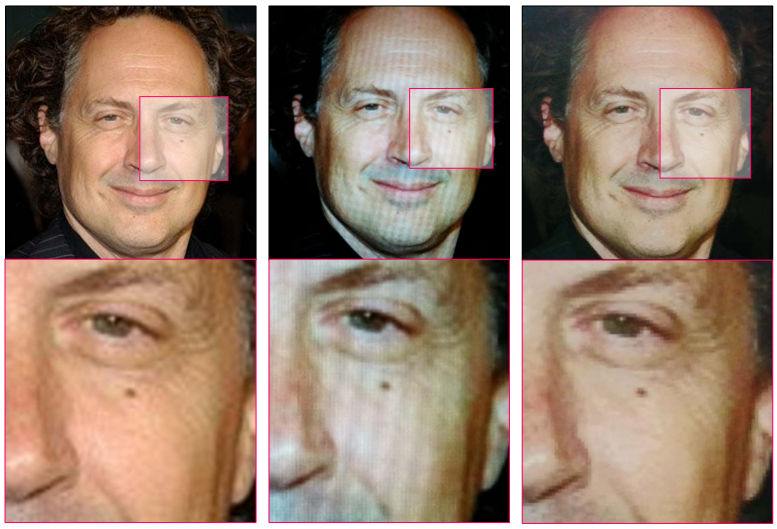

Figure 1. Examples of genuine faces and spoofing faces from the MSU USSA database [1]. Columns (a), (b), and (c) are the genuine face, display face, and printed photo face and their corresponding magnifying regions, respectively.

2.1.2. The deep-learning based methods

Recently, CNN-based methods, which have the powerful ability to learn deep representations from data, have attracted much research attention. Yang et al. [6] trained a CNN to learn deep representations for face anti-spoofing based on the AlexNet architecture [2]. After that, the feasibility of CNN in learning deep representation for biometrics, including face anti-spoofing, was further demonstrated by [35], and more CNN-based methods were increasingly reported [28, 36]. The models employed in these methods are simple feed-forward networks, which have limitations in capturing fine-grained spoofing information. To address this issue, Cai et al. [37] proposed a two-branch framework that utilizes reinforcement learning to extract and fuse local and global features for face anti-spoofing (FAS). Moreover, these previous works relied on binary labels, which are insufficient for providing the discriminative supervision signals needed to extract generalized features. Consequently,

pixel-wise supervision has been explored as an alternative [38–40]. Studies have shown that 2D pixel-wise supervision offers more informative and instructive signals for model training.

The above methods focus on the intra-domain scenario, but now the community has shifted its attention to the cross-domain generalization scenarios. When there are distribution shifts between training and testing data, the model's generalization performance is poor [36]. Thus, the domain generalization scenario is to improve model generalization capability without knowing the target domain data [41]. To solve this problem, meta-learning [9, 42, 43], disentangled representation learning [44–47], adversarial learning [41, 48, 49], and contrastive learning [48, 50, 51] have been proposed. In the unsupervised domain adaptation (UDA) scenario, the unlabeled target domain data is available to be used to adapt a model pre-trained on the source domain data, aiming to improve the model's performance in the target domain [52, 53]. When target domain data with labels is available, how to efficiently fine-tune the models studied in the paper focues on the few-shot cross-domain generalization scenario [10, 54, 55] and continual learning scenario [11].

These above methods mainly focus on the cross-domain generalization capability, but how to deal with the threats from adversarial attacks is not well studied. Although there are hybrid methods that combine NNs with handcrafted features, such as the LBP Network [56], such methods are also threatened by adversarial attacks [57]. There are other studies that have addressed the defense against adversarial examples in FAS. Bousnina et al. [58] suggested using eight data augmentation techniques to defend against differential evolution-based adversarial attacks, but this approach is specific to that type of attack. Deb et al. [59] introduced UniFAD, a framework aiming to combine digital and presentation attacks. However, UniFAD necessitates a substantial number of pre-prepared images for training an additional network. Our method explores classifiers other than NNs for FAS against adversarial attacks.

2.2. The tree-ensemble methods

The tree-ensemble methods are based on decision trees. The random decision forest is first proposed as a solution to the dilemma between performance and generalization of the decision tree [60]. It is later ameliorated to the Random Forest (RF) by introducing feature sampling and data bootstrapping by [61]. completely-random tree forest (CRF) has a mechanism that is much more "random" than RF since it splits the nodes randomly, regardless of any criteria [62]. Both the RF and the CRF would project original features into subspaces by sampling the original features. This reduces the dimensions of features to process, which facilitates the handling of high dimensional features [23]. The gradient tree boosting (GTB) methods introduce loss functions for training that have not been included in the RF and the CRF. The GTB models are trained by boosting the gradients of the loss. An effective way to implement GTB is proposed by [63], namely the XGBoost. The XGBoost provides a more flexible and powerful scheme that approximates non-differentiable loss functions by the first two terms of their Taylor expansion, so users are enabled to define arbitrary loss functions in their problems. The XGBoost has achieved superior performance among many GTB implementations. However, these tree-based methods lack layer-by-layer processing. Thus, Zhou et al. [23] proposed deep forest, which increases the model complexity by stacking tree models with depth.

3. Methodology

In this section, we first introduce the mechanism of deep forest. Then, we describe our solutions for improving deep forest with LBP features.

3.1. Deep forest

Deep forest, proposed by [23], can achieve competitive performance compared to CNN-based methods on several visual tasks reported by [23]. Similar to CNN models, deep forest has a number of layers, and each layer has N base RF classifiers. An RF classifier conducts binary classification: genuine or spoofing. The output of all RF in each layer is concatenated as the input of the next layer. When using deep the forest model [23] with image data, instead of using the raw pixel as the input of deep forest models, Zhou et al. [23] proposed the grain scanning mechanism (GSM) to learn input representation from raw pixels, which is depicted in Figure 2.

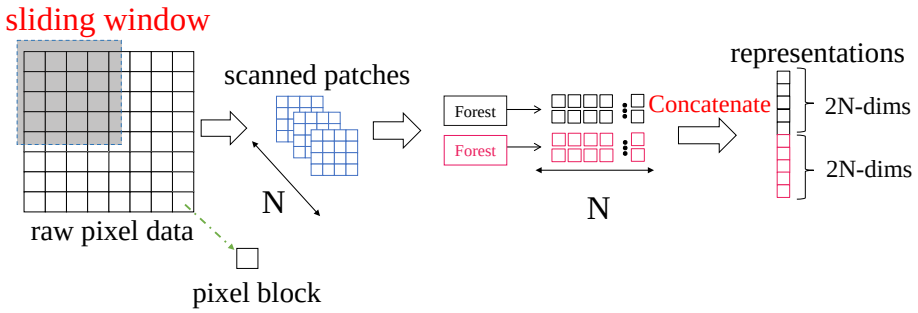

Figure 2. The illustration of how the GSM learns representations for local information [23]. First, a sliding window with a certain stride is used to scan raw pixels. Then, all the scanned patches are fed to forests, an RF (black) and a CRF (rose). Finally, all the output results from the forests will be concatenated as the representations of the raw pixel data. For full details about the GSM, please refer to [23].

RF is the basis of the GSM of deep forest and CRF offers another option for deep forest. By combining different types of forests, the diversity of representations learned by deep forest can be improved [23]. The XGBoost and other implementations of GTB can also be a basis in deep forest. Unlike CNNs, whose structures are fixed during the training process, the number of cascade levels of the deep forest model depends on the scale of the data and grows as the training proceeds. Once the output scores (accuracy, loss, etc.) begin to converge, the growth stops. Hence, the complexity of the model can be adaptively adjusted according to the scale of the database. This ensures that deep forest can maintain a satisfactory result even on a small-scale database [23]. More details about deep forest can be found in [23].

3.2. Why deep forest is robust against adversarial attacks

Existing NNs are mainly using backward propagation for optimization. A network's parameters can be defined as Θ, and a loss function $\mathcal{L}$, such as cross-entropy loss is used to optimize models.

According to the backward propagation, the update of Θ can be represented by

$$\Theta \leftarrow \Theta - \eta \nabla_{\Theta} \mathcal{L}, \tag{3.1}$$

where η represents the learning rate. The model is optimized along the direction of gradient descent. The key idea of launching adversarial examples is to generate adversarial noises that make models' gradients ascend. A representative method is the fast gradient sign method (FGSM) [69]. Assuming the input is x and its labels y, the adversarial example x^{adv} can be generated by

$$x^{adv} = x + sign(\nabla_x \mathcal{L}(\Theta, x, y)), \tag{3.2}$$

where $sign(\nabla_x \mathcal{L}(\Theta, x, y))$ is regarded as the adversarial noise, and $sign(x) = 1$ if $x > 0$ and $sign(x) = -1$ if $x < 0$. Adversarial noise contributes to an increase in model loss, thereby skewing the model toward incorrect predictions. In various methods, adversarial noise is primarily generated using gradient ascent techniques. However, deep forests operate differently, as their foundational components are decision trees. Unlike models that rely on gradient descent for optimization, decision trees are constructed through a greedy algorithm. This algorithm iteratively selects the optimal split at each node based on impurity measures, such as Gini impurity or entropy. Therefore, the adversarial noise hardly has a negative effect on the deep forest model. Furthermore, since the adversarial perturbations are applied to x in the RGB space, our proposed method transforms x into the LBP space, thereby further mitigating the associated risks. This approach will be discussed in greater detail in the following section.

3.3. The proposed deep forest model with LBP features

We re-devised the deep forest model by replacing the GSM with multi-scale LBP as the input processing. The LBP is selected for two reasons. First, LBP features cannot be reconstructed back to RGB pixel images, thus being helpful against adversarial attacks. Second, LBP is designed for texture description, which may be appropriate for the FAS problem. This section will first elaborate on how to use the LBP descriptors [35] to leverage texture information. Then, the proposed scheme integrating deep forest and the LBP features will be presented.

3.4. The LBP-based features for texture analysis

The LBP descriptor proposed by [64] is a gray-scale descriptor that is effective for texture description. By calculating the LBP values of the binary patterns for each pixel and accumulating the occurrences of them into histograms, the LBP features can be extracted to represent local texture information. The calculation of LBP can be described as

$$LBP_{P,R} = \sum_{n=1}^{P} \text{sgn}(r_n - r_c) \times 2^{n-1} \tag{3.3}$$

where $\text{sgn}(\cdot)$ takes the sign of the operand, r_c denotes the intensity value of the central pixel, and $r_n (n = 1, 2, ..., P)$ denotes the intensity values of P adjacent pixels distributed symmetrically at a circle of radius $R (R > 0)$. An image can be divided into several patches, and LBP histograms are calculated for each patch. Then, all the histograms can be concatenated into a feature vector to represent the image in the texture field. To fully exploit the color information, color LBP features will be employed by

referring to [34] in this paper. The color LBP features are to extract LBP features from each component individually of the color space (e.g., red, blue, and green in the RGB space or hue, saturation, and value in the HSV space) and the obtained results will be concatenated into a feature vector [34]. These features based on LBP descriptors are to be called LBP features in this paper.

The GSM learns the representations of local information from adjacent pixels within a certain window, and similarly, the extraction of LBP-based features also considers the local information. On the other hand, the significant contrast between employing the GSM [23] (illustrated in Figure 2) and the LBP features lies in the representation extraction. The GSM constructs representations by learning from data while the LBP features construct representations with the domain knowledge of a researcher.

3.5. The proposed multi-scale representations

First, we propose to use the multi-scale LBP descriptor to construct the multi-scale representations. Taking multi-scales into account is important because the image samples are from practical capturing conditions and there are variations in the textural disparities. For example, although both Figure 1(b) and (c) are spoofing faces, they are captured under different conditions, i.e., different devices, different circumstances, etc., so they show different texture appearances in both patterns and scales. Therefore, different scales of local information should be taken into consideration. As is illustrated in Figure 3(a), the multi-grained scanning mechanism (MGSM) [23] is used to learn representations from data on multiple scales. By changing the size of the sliding windows and conducting the GSM, relationships of the pixels on different scales will be learned, and local information on different scales can be leveraged [23]. On the other hand, Figure 3(b) illustrates our proposed scheme. In the proposed scheme, there is a sliding window for scanning patches of pixels, and $\mathrm{LBP}^{u2}_{P,R}$ descriptors [64] are used to obtain LBP features. By changing the parameters P and R, representations on different scales can be obtained. To utilize color information, the color LBP features will be adopted in the proposed scheme to construct representations on different color channels and scales according to [34]. One of the differences between the MGSM and our proposed scheme in constructing multi-scale representations lies in the selection of sliding windows. With the MGSM, windows of different sizes are needed to learn multi-scale representations, while multi-scale representations based on LBP features can be obtained with a fixed-size window. This is because the representations on a certain scale learned by the GSM only depend on the size of the window; while, in the exploitations of LBP descriptors, the representation in a certain scale can also be determined by certain parameters of LBP descriptors, i.e., P and R. Multiple sizes of windows are not adopted in this paper for the consideration that when small-size windows are used to extract LBP histograms, many of the bins are empty, and the obtained features will be high-dimension and sparse, i.e., less informative.

Second, instead of concatenating all the representations on these three scales to construct a feature vector, as performed in some traditional methods [1, 34], we propose a circular cascading strategy based on the vanilla cascade strategy in [23]. This strategy is shown in Figure 4. The n-th layer will be identified as L_n. Representations on the three scales are denoted by S^1_{LBP}, S^2_{LBP}, and S^3_{LBP}, respectively. They will be individually fused with the output of each layer of deep forest, and each layer will focus on the representations on a certain scale. The S^1_{LBP} will be fed to the first layer of deep forest and fused with the output of L_1. Then, the representation S^2_{LBP} will be fused with the output from L_1 and become

the input of L_2. S^3_{LBP} and L_3 will do the same. It should be noted that this cascading process is circular. For instance, in the next circle, the S^1_{LBP} is concatenated in the L_4. In the k-th circle, the S^1_{LBP} will be concatenated in the $L_{3k-2}, k \in \mathbb{N}$. Actually, the options of the scales and cascade strategies are flexible according to tasks.

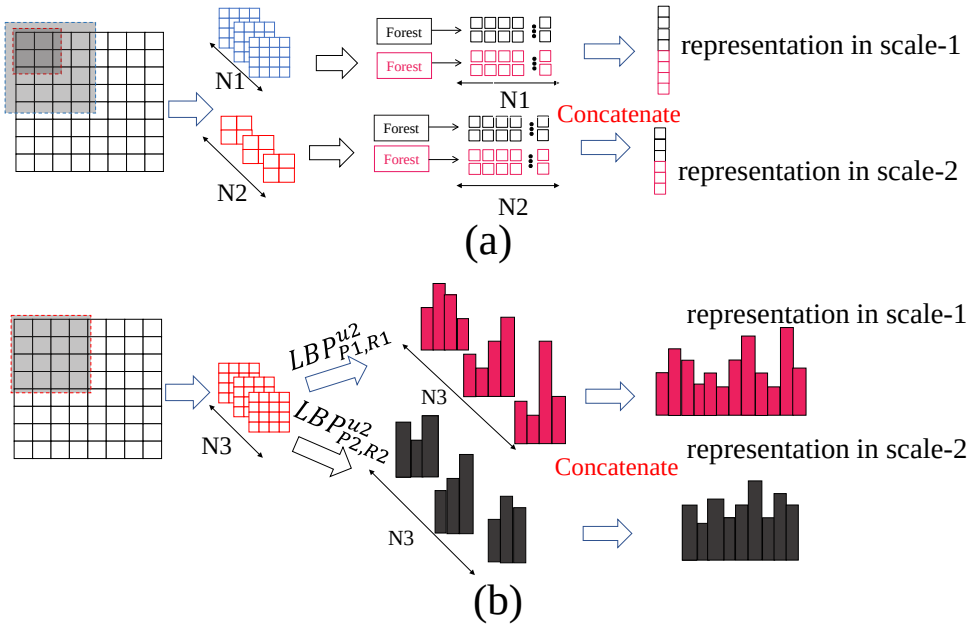

Figure 3. Illustrations of constructing multi-scale representations. (a) and (b) illustrate how the MGSM and the proposed scheme construct representations on multi-scales, respectively.

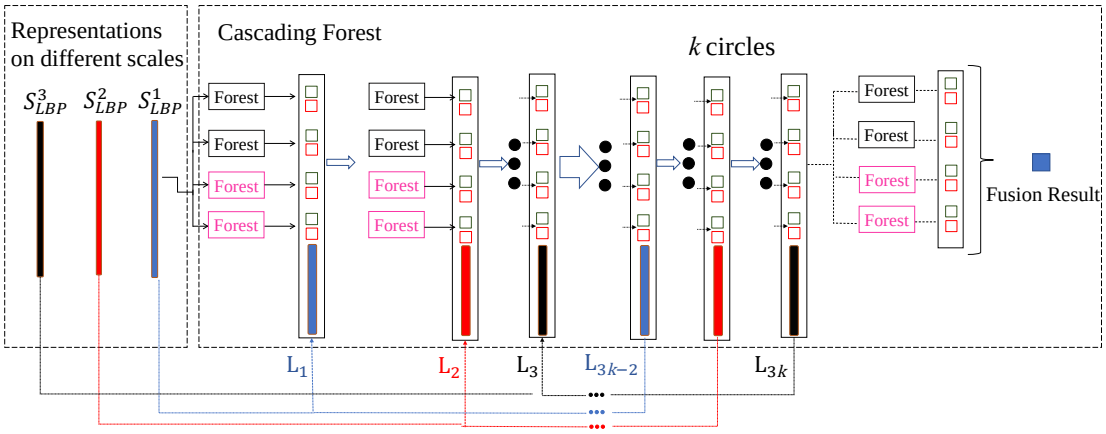

Figure 4. The procedure of deep forest learning with multi-scale representations. The left part contains the LBP representations on three different scales, denoted by S^1_{LBP}, S^2_{LBP}, and S^3_{LBP}, respectively. The right part illustrates the cascading strategy. The black and red boxes are output results of each forest from the previous layer. They will be concatenated with features on different scales in different layers as the input of their next layers.

4. Experiments

In this section, a brief introduction to four databases, on which the experiments are conducted, will be first given. Then, the details about the settings of the experiments are shown. Finally, the experimental results are presented and discussed.

4.1. Databases

In our experiment, four representative databases have been employed. Two are benchmark databases, CASIA FASD [65] and IDIAP REPLAY-ATTACK [27], and another two are newly-published databases, the ROSE-YOUTU LIVENESS database [52] and MSU USSA database [1]. The IDIAP, CASIA, and ROSE-YOUTU databases consist of videos, covering replay attacks, display attacks, and print attacks. The MSU database only contains images, i.e., only including display attacks and print attacks. More specifically, the scales of each database are summarized below.

The IDIAP REPLAY-ATTACK database [27] constitutes about 50 subjects. There are 60 videos of genuine faces and 300 videos of fake faces in the training set. In the testing set, there are 80 videos of genuine faces and 400 videos of spoofing faces.

The CASIA database [65] consists of 600 videos from 50 subjects, 20 subjects for the training set and 30 subjects for the testing set. For each subject, there are 3 videos of genuine faces and 9 videos of spoofing faces.

Figure 5. Examples of PA. (a): Display attack: The face is in a digital display screen [27]. (b) Replay attack [65]: The face is in a video. (c) Print attack: The face is in a print photo [1]. (d) Print attack: The face is in a print photo that is tailored [52].

The ROSE-YOUTU LIVENESS database [52] contains 10 and 12 subjects in the training set and the testing set, respectively. For each subject, there are 180 videos, consisting of various types of attack and light conditions. This database is the latest database concerning PA, and there is a tailored print attack. As can be seen in Figure 5, the background is not included in the recapturing process, making this database more challenging.

The MSU USSA database [1] includes 1000 genuine faces (about 1000 subjects) and about 6000 spoof face images. There is no division between the training set and the testing set, so a 5-fold validation protocol is used to evaluate the performance of the FAS methods.

4.2. Experimental setups

In the first place, it should be highlighted that some data preprocessing has been performed in our experiments. When conducting experiments on the MSU and IDIAP databases, the whole image frames are taken as the inputs for making full use of information. This is because the PA places the spoof media near the cameras to achieve high recapturing quality, and the "background" is also recaptured (as shown in (a) and (c) of Figure 5). The recaptured background provides useful information which is beneficial to spoof face detection. However, in the CASIA database and the ROSE-YOUTU database, the PA is far away from the cameras and hence the background is not included in the recapturing process (as shown in (b) and (d) of Figure 5). Under this circumstance, if the whole frame is used as the input, unnecessary interference (from the genuine background) will be introduced. Therefore, the Viola-Jones method [66] is utilized to detect the faces in frames from the ROSE-YOUTU and CASIA databases. The detected faces are cropped and employed as the inputs in the experiments. Then the resolution of all the inputs (i.e., whole frames from the IDIAP database and MSU database or cropped face regions from the CASIA database and ROSE-YOUTU database) is normalized to 128×128 pixels for a trade-off between the computational complexity and performance by referring to the prior works [1].

Second, the settings of the experiments are elaborated. In [23], three square sliding windows of different sizes are employed to evaluate the performance of deep forest. By referring to this, three scales of windows, 16, 32, and 64 pixels, with the strides of 8, 16, and 32 pixels, respectively, are used for the MGSM in this paper. The obtained representations on these three scales will be denoted by S^1_{GSM}, S^2_{GSM}, and S^3_{GSM}, respectively. In our proposed scheme, the size of the sliding window is fixed at 32 pixels and the stride is 16 pixels. For each image of 128×128, there will be 7×7 overlapped sub-patches in total. Three $\mathrm{LBP}^{u2}_{P,R}$ descriptors [64], $\mathrm{LBP}^{u2}_{8,1}$, $\mathrm{LBP}^{u2}_{16,2}$, and $\mathrm{LBP}^{u2}_{24,3}$, are utilized to construct representations on three scales, and the obtained representations are referred to as S^1_{LBP}, S^2_{LBP}, and S^3_{LBP}, respectively. Also, color LBP features in HSV and YCbCr spaces are considered in this paper. That is to extract features in each separate channel of an image. For one patch, the feature lengths of color (RGB, HSV, YCbCr) $\mathrm{LBP}^{u2}_{8,1}$, $\mathrm{LBP}^{u2}_{16,2}$, and $\mathrm{LBP}^{u2}_{24,3}$ are 59×3, 59×3, and 59×3, respectively. Since there are 49 sub-patches for each image, the lengths of the final S^1_{LBP}, S^2_{LBP}, and S^3_{LBP} will be $49 \times 59 \times 3$, $49 \times 243 \times 3$, $49 \times 555 \times 3$ respectively. During the cascading operation, S^1_{LBP} / S^1_{GSM} will be fused with L_1, S^2_{LBP} / S^2_{GSM} with L_2 and S^3_{LBP} / S^3_{GSM} with L_3. This process continues circularly until the training process terminates. This process will stop automatically when accuracies converge for several rounds. As for the setting of forests utilized in deep forest, four RFs and four CRFs are employed and there are 500 trees in each forest by referring to [23]. These are implemented with the package of the gcForest (https://github.com/kingfengji/gcForest) with default settings of the forests.

For more details of the mechanism of deep forest, please refer to [23].

4.3. Experimental results

4.3.1. Proof of the weakness of neural network models

To show the effectiveness of our proposed model against adversarial noise, we show the performance degradation in Table 1. In Table 1, we compared our proposed method with benchmark CNNs (AlexNet [2], VGG-16 [67], ResNet-18 [68], CDCN [8]), vision transformer (ViT-16B) [4], and ViT with S-Adapter [10]. We utilized the fast gradient sign method (FGSM) [69] to generate adversarial examples from the IDIAP dataset based on the AlexNet model [2] with the training data. The ϵ is the coefficient parameter of the FGSM method that controls the magnitude of the adversarial noises. The bigger the ϵ, the more adversarial noises are added to the image input. As shown in Table 1, we can see that the adversarial noises are transferable as the adversarial noises are based on AlexNet but all neural network models are also vulnerable, as shown by the increasing EER. By contrast, our proposed method with different color space input remains steady in its performance.

Table 1. Model performance of EER against the adversarial attack with the IDIAP-REPLAY ATTACK dataset.

Model	Year	$\epsilon = 0.0$	$\epsilon = 0.1$	$\epsilon = 0.25$	$\epsilon = 0.5$
AlexNet [2]	2012	0.00	30.71	38.04	49.86
VGG-16 [67]	2015	0.00	25.23	35.45	47.28
ResNet-18 [68]	2016	0.00	24.67	32.75	39.73
CDCN [8]	2020	0.012	16.45	21.23	35.46
ViT-16B [4]	2021	0.00	14.02	17.76	37.21
ViT with S-Adapter [10]	2024	0.002	8.76	13.29	37.42
Our proposed (RGB)	–	0.00	0.089	1.34	2.56
Our proposed (HSV)	–	0.052	0.064	1.01	2.86
Our proposed (YCBCR)	–	**0.00**	**0.035**	**0.65**	**2.23**

4.3.2. Comparisons between multi-scale representations

Table 2 provides the experimental results of the GSM and of the proposed scheme in terms of EER. From Table 2, by integrating the LBP features (RGB) with deep forest, the EER on the MSU, IDIAP, and CASIA databases are reduced from 4.84 to 4.17%, from 1.02 to 0%, and from 14.50 to 11.82%, respectively. These results suggest that LBP-based features are more competent in exploiting texture information to represent the degradation of the spoofing faces than the GSM. Furthermore, across different color spaces, the performances of LBP features in HSV and YCbCr color spaces are generally better than those in the RGB color space. This is because the change of illuminance should not interfere with chrominance information, which is crucial in color texture methods, and the HSV space and YCbCr space separate prime components of illumination and chrominance. However, the RGB space has high correlations in the three components, and slight variance of illumination by altering the R, G, and B may result in unexpected change in the chrominance, making the feature less effective [34].

Table 2. Comparisons between two implementations of multi-scale representations on the MSU USSA database, IDIAP database, and CASIA database. Performance is evaluated by EER (%).

Multi-scale representations	Database		
	MSU	IDIAP	CASIA
GSM (RGB) [23]	4.84	1.02	14.50
proposed (RGB)	4.17	**0.00**	11.82
proposed (HSV)	2.14	0.052	**8.73**
proposed (YCBCR)	**1.56**	**0.00**	9.66

To further probe into the effectiveness of the proposed scheme, curves of the training accuracy outputted by each layer are drawn and shown in Figure 6. An upward trend of the accuracy results can be seen. It goes up along with the growth of the structure. There are limited improvements in the curve over different layers with the GSM, which indicates that the GSM is not able to capture the texture information of the spoofing cues over different scales efficiently. Meanwhile, despite inferior accuracies in the first two layers, the accuracies of the proposed scheme (RGB, HSV, YCbCr) finally outperform that of the GSM. Moreover, the trend of the curves indicates that the cascading strategy enables LBP features to be re-represented. For instance, S^1_{LBP} is fed to the layers L_1, L_4, and L_7, and the outputted accuracies get improved. In layer L_1, the deep forest model learns from S^1_{LBP}, representations on a small scale. Then, after L_2, and L_3, where the model has perceived more information from representations on larger scales, the model leads to a better understanding of the distortion on different scales.

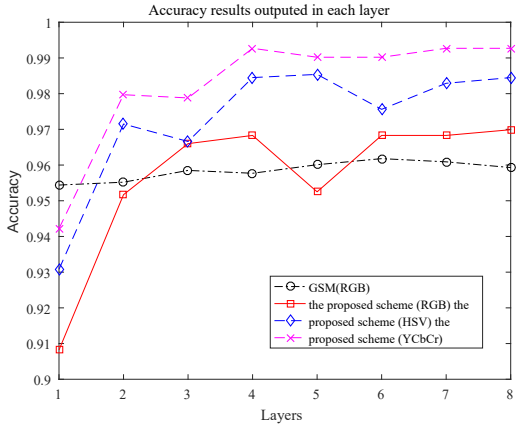

Figure 6. The convergence curves of experiments on the MSU database. An average of the results of the five validations is taken. The x-axis refers to the number of the cascade layer that increases along with the training process. The y-axis refers to the testing accuracy of the output of each layer.

4.3.3. Comparisons with state-of-the-art approaches

Tables 3 and 4 provide results of comparisons between the proposed scheme and the state-of-the-art approaches. From Table 3, the proposed scheme with simple LBP features is

demonstrated to be highly competitive. First, on the CASIA database, the proposed scheme (HSV) achieves 8.73% EER. Although this result is inferior to the results of some CNN-based methods, the patch-based CNN (4.44%) [28], the depth-based CNN (2.85%) [28], and fine-tuned AlexNet (6.1%) [6], it is better than some LSTM-based method with 22.40 and 14.60% EERs presented in [70]. It is worth mentioning that, among the traditional methods (using the SVM classifiers with handcrafted features), particularly among LBP-based methods, the co-occurrence of adjacent local binary patterns (CoALBP) method [26] has achieved state-of-the-art performance [52]. Experiments on the CASIA database show that the proposed scheme with LBP features has achieved a better result (9.66%) than CoALBP (10.0%) in the YCbCr space. Moreover, experimental results on the ROSE-YOUTU database [52], a more diverse and challenging database, are presented in the last column in Table 3. The results show that the CoALBP, which performs well on the IDIAP database (3.7% in HSV and 1.4% in YCbCr) and CASIA database (5.0% in HSV and 10.0% in YCbCr), drops dramatically (16.4% in HSV and 17.7% in YCbCr) [52]. However, the proposed scheme, which is also related to the LBP, achieves 10.9% (HSV) and 11.9% (YCbCr). Furthermore, from Table 3, the proposed scheme achieves 0% (YCbCr) on the IDIAP REPLAY-ATTACK database, which is better than the results of all the presented CNN-based methods and CoALBP. On the CASIA and ROSE-YOUTU datasets, deep forest based on LBP is inferior to the CNN and ViT methods. After we observed the data, we found that the ROSE-YOUTU and CASIA datasets are more complex than IDIAP as they contain more lighting settings. Based on the results in Table 1, we can conclude that our proposed method is recommended in the scenario when the lighting settings are simple and consistent if considering the robustness of adversarial attacks. A more recommended solution is to ensemble our proposed method with an NN-based method in a system, and such an ensemble solution can be studied in the future.

Table 3. Comparisons between the proposed scheme and state-of-the-art approaches on the IDIAP database, CASIA database, and ROSE-YOUTU database, which are in terms of EER (%).

Method	Year	IDIAP	CASIA	ROSE-YOUTU
LBP-TOP [32]	2012	7.9	–	–
CoALBP (HSV) [34]	2017	3.7	5.5	16.4
CoALBP (YCbCr) [34]	2017	1.4	10.0	17.7
Fine-tuned AlexNet [6]	2014	6.1	7.4	8.0
CNN+Conv-LSTM [70]	2018	5.12	22.40	–
CNN+LSTM [70]	2018	1.28	14.60	–
Patch-based CNN [28]	2018	2.5	4.44	–
Depth-based CNN [28]	2018	0.86	2.85	–
CDCN [8]	2020	0.012	1.02	1.78
ViT with S-Adapter [10]	2024	0.002	0.89	1.24
RF (HSV)	–	3.44	13.1	15.7
RF (YCbCr)	–	2.45	12.2	14.5
proposed (HSV)	–	0.052	8.73	10.9
proposed (YCbCr)	–	0	9.66	11.9

The experimental results on the MSU USSA database, in terms of the EER and the half-total error rate (HTER), are provided in Table 4. According to Table 4, methods based on the CNN and ViT [8, 10, 28] achieve satisfactory results in both EER and HTER below 1% on the MSU database. Our proposed scheme in YCbCr space achieves 1.56% EER and 1.33% HTER, which is competitive also, as the depth-based CNN [28] achieves 2.62% EER and 2.22% HTER. In summary, taking Tables 1, 3, and 4 together, our method can be considered as a backup solution when facing threats from adversarial noises. Our solution is suitable when the environments are relatively consistent.

Table 4. Performance in terms of EER (%) and HTER (%) on the MSU USSA database. The results are obtained according to the 5-fold validation protocol in [1].

Method	EER	HTER
Patel et al. [1]	3.84	-
Patch-based CNN [28]	0.55 ± 0.26	0.41 ± 0.32
Depth-based CNN [28]	2.62 ± 0.73	2.22 ± 0.66
CDCN [8]	0.34 ± 0.26	0.23 ± 0.15
ViT with S-Adapter [10]	0.47 ± 0.25	0.35 ± 0.22
proposed (HSV)	2.14 ± 0.58	1.98 ± 0.58
proposed (YCbCr)	1.56 ± 0.61	1.33 ± 0.51

4.3.4. Comparisons of different numbers of trees in each forest

In the above experiments, we follow [23] and adopt 500 trees in each forest. In a certain range, the more trees in a forest, the better the performance. However, too many trees in a forest would introduce heavy computational costs. In [71], it is suggested that a trade-off between performance and computational costs can be achieved when the number of trees in a forest is in the range of 64 to 128. There are no significant performance gains when the number of trees increases to 512, 1024, 2048, or other larger numbers. Experimental results in Table 4 show that when the number of trees is smaller than 500, the performance does not necessarily drop. This observation coincides with the conclusion in [71].

4.3.5. Computational cost

We evaluated the inference computation of the deep forest model. In Table 5, we compare our deep forest with prevalent cornerstone models: VGG-16, ResNet-18, and ViT-B 16 on our server. Corresponding to Table 6, we evaluate our deep forest model with 64, 128, 256, and 500 trees in each base forest model. The machine used for evaluation has an Intel(R) Xeon(R) CPU E5-2650 v2 (32 core) and an NVIDIA RTX A5000 GPU. Since deep forest currently only runs on the CPU, GPU results for deep forest are not available (N.A.). While the neural network models can achieve very fast inference times of less than 100ms per iteration, the GPU A5000 used is expensive, costing nearly $2000, and may not be available on some embedded devices. Notably, when the number of trees is set to 64, the inference time for deep forest is smaller than that of ResNet-18, demonstrating its efficiency.

Table 5. Inference time of different models (ms). The batch size is 16. The used GPU is NVIDIA RTX A5000, and the CPU is Intel(R) Xeon(R) CPU E5-2650 v2 (32 core).

Computation cost	CPU (ms)	GPU (ms)
ResNet-18	269.2	16.2
VGG-16	2226.9	30.8
ViT-B16	2118.4	57.1
Deep forest (64)	245.4	N.A.
Deep forest (128)	374.5	N.A.
Deep forest (256)	529.2	N.A.
Deep forest (500)	1070.2	N.A.

Table 6. Performance (EER %) of different numbers of trees in each forest.

Dataset	Number of trees	64	128	256	500
CASIA	HSV	8.62	8.59	8.67	8.73
	YCbCr	9.53	9.54	9.61	9.66
IDIAP	HSV	0.054	0.047	0.048	0.052
	YCbCr	0.026	0.017	0.023	0
MSU	HSV	1.99	1.96	2.22	2.14
	YCbCr	1.26	1.28	1.42	1.51
ROSE-YOUTU	HSV	10.4	10.4	10.7	10.9
	YCbCr	11.4	11.5	11.3	11.9

5. Conclusions

5.1. Summary

Given the concern on the adversarial attack, in this paper, we propose to utilize the deep forest method [23] for the problem of the FAS. To the best of our knowledge, this is the first attempt to introduce deep forest into the FAS problem. Inspired by works related to texture analysis, we re-devise the construction of multi-scale representations by integrating LBP descriptors with deep forest learning scheme. Our proposed scheme has achieved better results than the original GSM proposed by [23]. Furthermore, compared with the state-of-the-art approaches, competitive results have been achieved on several benchmark databases by the proposed scheme. For example, 0% EER is achieved on the IDIAP dataset. This indicates the effectiveness and competitiveness of our proposed scheme. Hence, our method could offer a competitive option to those who would like to improve the security of their systems by fusing diverse approaches in their schemes at the system level. Moreover, there have been a limited number of research works that exploit deep forest on practical problems. This paper could serve as an important reference for researchers who want to explore methods beyond the CNN-based schemes.

5.2. Limitation and future work

Admittedly, the results of our approach do not look as attractive as some NN-based methods, such as CDCN [8] and S-Adapter [10]. Our proposed method is recommended for scenarios with simple and consistent lighting conditions, particularly when considering robustness against adversarial attacks. A more robust solution would be to combine our proposed method with a vision transformer (ViT) or neural network-based method in an ensemble system, which could be explored in future research. Also, various efforts can be made to improve the overall performance, such as investigating more cascading strategies and feature extraction methods. In this work, the LBP is utilized because it is common in the field of the FAS problem and it is relatively simple for us to implement in deep forest. However, the LBP is designed by researchers in computer vision society based on their domain knowledge. Such knowledge may not be fully applicable to the FAS problem. Some novel methods of binary descriptors have raised our strong interest and given us significant references [72–75]. Designed in a more intellectual way, they can learn features from data and are less dependent on people's knowledge. Hopefully, we can achieve better results by using these methods.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was supported by the NTU-PKU Joint Research Institute, a collaboration between the Nanyang Technological University (NTU) and Peking University (PKU) that was sponsored by a donation from the Ng Teng Fong Charitable Foundation. Also, the Science and Technology Foundation of Guangzhou Huangpu Development District provided funding for this research work under Grant No. 2022GH15.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. K. Patel, H. Han, A. K. Jain, Secure face unlock: Spoof detection on smartphones, *IEEE Trans. Inf. Forensics Secur.*, **11** (2016), 2268–2283. https://doi.org/10.1109/TIFS.2016.2578288
2. A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM*, **60** (2017), 84–90. https://doi.org/10.1145/3065386
3. Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, **2** (2014), 1988–1996.
4. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al., An image is worth 16×16 words: Transformers for image recognition at scale, preprint, arXiv:2010.11929.

5. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, et al. , Swin transformer: Hierarchical vision transformer using shifted windows, in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2021), 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986

6. J. Yang, Z. Lei, S. Z. Li, Learn convolutional neural network for face anti-spoofing, preprint, arXiv:1408.5601.

7. Z. Xu, S. Li, W. Deng, Learning temporal features using LSTM-CNN architecture for face anti-spoofing, in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, (2016), 141–145. https://doi.org/10.1109/ACPR.2015.7486482

8. Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, et al., Searching central difference convolutional networks for face anti-spoofing, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), 5294–5304. https://doi.org/10.1109/CVPR42600.2020.00534

9. R. Cai, Z. Li, R. Wan, H. Li, Y. Hu, A. C. Kot, Learning meta pattern for face anti-spoofing, *IEEE Trans. Inf. Forensics Secur.*, **17** (2022), 1201–1213. https://doi.org/10.1109/TIFS.2022.3158551

10. R. Cai, Z. Yu, C. Kong, H. Li, C. Chen, Y. Hu, et al., S-adapter: Generalizing vision transformer for face anti-spoofing with statistical tokens, *IEEE Trans. Inf. Forensics Secur.*, **19** (2024), 8385–8397. https://doi.org/10.1109/TIFS.2024.3420699

11. R. Cai, Y. Cui, Z. Li, Z. Yu, H. Li, Y. Hu, et al., Rehearsal-free domain continual face anti-spoofing: Generalize more and forget less, in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2023), 8003–8014. https://doi.org/10.1109/ICCV51070.2023.00738

12. A. Liu, Z. Tan, Z. Yu, C. Zhao, J. Wan, Y. Liang, et al., FM-ViT: Flexible modal vision transformers for face anti-spoofing, *IEEE Trans. Inf. Forensics Secur.*, **18** (2023), 4775–4786. https://doi.org/10.1109/TIFS.2023.3296330

13. C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, et al., Intriguing properties of neural networks, preprint, arXiv:1312.6199.

14. A. Aldahdooh, W. Hamidouche, O. Deforges, Reveal of vision transformers robustness against adversarial attacks, preprint, arXiv:2106.03734.

15. A. Kurakin, I. J. Goodfellow, S. Bengio, Adversarial examples in the physical world, preprint, arXiv:1607.02533.

16. K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, et al., Robust physical-world attacks on deep learning visual classification, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 1625–1634. https://doi.org/10.1109/CVPR.2018.00175

17. M. Sharif, S. Bhagavatula, L. Bauer, 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), 1528–1540. https://doi.org/10.1145/2976749.2978392

18. Y. Liu, X. Chen, C. Liu, D. Song, Delving into transferable adversarial examples and black-box attacks, preprint, arXiv:1611.02770.

19. W. Ma, Y. Li, X. Jia, W. Xu, Transferable adversarial attack for both vision transformers and convolutional networks via momentum integrated gradients, in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2023), 4607–4616. https://doi.org/10.1109/ICCV51070.2023.00427

20. P. Russu, A. Demontis, B. Biggio, G. Fumera, F. Roli, Secure kernel machines against evasion attacks, in *Proceedings of the 2016 ACM workshop on artificial intelligence and security*, (2016), 59–69. https://doi.org/10.1145/2996758.2996771

21. F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, P. McDaniel, The space of transferable adversarial examples, preprint, arXiv:1704.03453.

22. F. Yang, Z. Chen, Using randomness to improve robustness of machine-learning models against evasion attacks, preprint, arXiv:1808.03601.

23. Z. Zhou, J. Feng, Deep forest: Towards an alternative to deep neural networks, in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, (2017), 3553–3559. https://doi.org/10.24963/ijcai.2017/497

24. J. Maatta, A. Hadid, M. Pietikäinen, Face spoofing detection from single images using micro-texture analysis, in *2011 International Joint Conference on Biometrics (IJCB)*, (2011), 1–7. https://doi.org/10.1109/IJCB.2011.6117510

25. J. Yang, Z. Lei, S. Liao, S. Z. Li, Face liveness detection with component dependent descriptor, in *2013 International Conference on Biometrics (ICB)*, (2013), 1–6. https://doi.org/10.1109/ICB.2013.6612955

26. R. Nosaka, Y. Ohkawa, K. Fukui, Feature extraction based on co-occurrence of adjacent local binary patterns, in *Advances in Image and Video Technology*, **7088** (2011), 82–91. https://doi.org/10.1007/978-3-642-25346-1_8

27. I. Chingovska, A. Anjos, S. Marcel, On the effectiveness of local binary patterns in face anti-spoofing, in *2012 BIOSIG-Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG)*, (2012), 1–7.

28. Y. Atoum, Y. Liu, A. Jourabloo, X. Liu, Face anti-spoofing using patch and depth-based CNNs, in *2017 IEEE International Joint Conference on Biometrics (IJCB)*, (2017), 319–328. https://doi.org/10.1109/BTAS.2017.8272713

29. X. Tan, Y. Li, J. Liu, L. Jiang, Face liveness detection from a single image with sparse low rank bilinear discriminative model, in *Computer Vision-ECCV 2010*, **6316** (2010), 504–517. https://doi.org/10.1007/978-3-642-15567-3_37

30. D. Gragnaniello, G. Poggi, C. Sansone, L. Verdoliva, An investigation of local descriptors for biometric spoofing detection, *IEEE Trans. Inf. Forensics Secur.*, **10** (2015), 849–863. https://doi.org/10.1109/TIFS.2015.2404294

31. Z. Boulkenafet, J. Komulainen, A. Hadid, Face antispoofing using speeded-up robust features and fisher vector encoding, *IEEE Signal Process. Lett.*, **24** (2017), 141–145. https://doi.org/10.1109/LSP.2016.2630740

32. T. D. F. Pereira, A. Anjos, J. M. D. Martino, S. Marcel, LBP-TOP based countermeasure against face spoofing attacks, in *Computer Vision-ACCV 2012 Workshops*, **7728** (2012), 121–132. https://doi.org/10.1007/978-3-642-37410-4_11

33. S. R. Arashloo, J. Kittler, W. Christmas, Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features, *IEEE Trans. Inf. Forensics Secur.*, **10** (2015), 2396–2407. https://doi.org/10.1109/TIFS.2015.2458700

34. Z. Boulkenafet, J. Komulainen, A. Hadid, Face spoofing detection using colour texture analysis, *IEEE Trans. Inf. Forensics Secur.*, **11** (2016), 1818–1830. https://doi.org/10.1109/TIFS.2016.2555286

35. D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcão, et al., Deep representations for iris, face, and fingerprint spoofing detection, *IEEE Trans. Inf. Forensics Secur.*, **10** (2015), 864–879. https://doi.org/10.1109/TIFS.2015.2398817

36. H. Li, P. He, S. Wang, A. Rocha, X. Jiang, A. C. Kot, Learning generalized deep feature representation for face anti-spoofing, *IEEE Trans. Inf. Forensics Secur.*, **13** (2018), 2639–2652. https://doi.org/10.1109/TIFS.2018.2825949

37. R. Cai, H. Li, S. Wang, C. Chen, A. C. Kot, DRL-FAS: A novel framework based on deep reinforcement learning for face anti-spoofing, *IEEE Trans. Inf. Forensics Secur.*, **16** (2020), 937–951. https://doi.org/10.1109/TIFS.2020.3026553

38. W. Sun, Y. Song, C. Chen, J. Huang, A. C. Kot, Face spoofing detection based on local ternary label supervision in fully convolutional networks, *IEEE Trans. Inf. Forensics Secur.*, **15** (2020), 3181–3196. https://doi.org/10.1109/TIFS.2020.2985530

39. A. George, S. Marcel, Deep pixel-wise binary supervision for face presentation attack detection, preprint, arXiv:1907.04047.

40. Z. Yu, X. Li, J. Shi, Z. Xia, G. Zhao, Revisiting pixel-wise supervision for face anti-spoofing, *IEEE Trans. Biom. Behav. Identity Sci.*, **3** (2021), 285–295. https://doi.org/10.1109/TBIOM.2021.3065526

41. R. Shao, X. Lan, J. Li, P. C. Yuen, Multi-adversarial discriminative deep domain generalization for face presentation attack detection, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2019), 10015–10023. https://doi.org/10.1109/CVPR.2019.01026

42. Y. Qin, Z. Yu, L. Yan, Z. Wang, C. Zhao, Z. Lei, Meta-teacher for face anti-spoofing, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2022), 6311–6326. https://doi.org/10.1109/TPAMI.2021.3091167

43. Z. Yu, J. Wan, Y. Qin, X. Li, S. Z. Li, G. Zhao, NAS-FAS: Static-dynamic central difference network search for face anti-spoofing, *IEEE Trans. Pattern Anal. Mach. Intell.*, **43** (2021), 3005–3023. https://doi.org/10.1109/tpami.2020.3036338

44. Y. Liu, J. Stehouwer, X. Liu, On disentangling spoof trace for generic face anti-spoofing, in *Computer Vision-ECCV 2020*, **12363** (2020), 406–422. https://doi.org/10.1007/978-3-030-58523-5_24

45. K. Zhang, T. Yao, J. Zhang, Y. Tai, S. Ding, J. Li, et al., Face anti-spoofing via disentangled representation learning, in *Computer Vision-ECCV 2020*, **12364** (2020), 641–657. https://doi.org/10.1007/978-3-030-58529-7_38

46. H. Wu, D. Zeng, Y. Hu, H. Shi, T. Mei, Dual spoof disentanglement generation for face anti-spoofing with depth uncertainty learning, *IEEE Trans. Circuits Syst. Video Technol.*, **32** (2022), 4626–4638. https://doi.org/10.1109/TCSVT.2021.3133620

47. W. Yan, Y. Zeng, H. Hu, Domain adversarial disentanglement network with cross-domain synthesis for generalized face anti-spoofing, *IEEE Trans. Circuits Syst. Video Technol.*, **32** (2022), 7033–7046. https://doi.org/10.1109/TCSVT.2022.3178723

48. Y. Jia, J. Zhang, S. Shan, X. Chen, Single-side domain generalization for face anti-spoofing, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), 8481–8490. https://doi.org/10.1109/CVPR42600.2020.00851

49. Z. Wang, Z. Wang, Z. Yu, W. Deng, J. Li, T. Gao, et al., Domain Generalization via Shuffled Style Assembly for Face Anti-Spoofing, in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2022), 4113–4123. https://doi.org/10.1109/CVPR52688.2022.00409

50. A. Liu, C. Zhao, Z. Yu, J. Wan, A. Su, X. Liu, et al., Contrastive context-aware learning for 3D high-fidelity mask face presentation attack detection, *IEEE Trans. Inf. Forensics Secur.*, **17** (2022), 2497–2507. https://doi.org/10.1109/TIFS.2022.3188149

51. A. George, S. Marcel, Learning one class representations for face presentation attack detection using multi-channel convolutional neural networks, *IEEE Trans. Inf. Forensics Secur.*, **16** (2020), 361–375. https://doi.org/10.1109/TIFS.2020.3013214

52. H. Li, W. Li, H. Cao, S. Wang, F. Huang, A. C. Kot, Unsupervised Domain Adaptation for Face Anti-Spoofing, *IEEE Trans. Inf. Forensics Secur.*, **13** (2018), 1794–1809. https://doi.org/10.1109/TIFS.2018.2801312

53. Y. Liu, Y. Chen, W. Dai, M. Gou, C. Huang, H. Xiong, Source-free domain adaptation with contrastive domain alignment and self-supervised exploration for face anti-spoofing, in *Computer Vision-ECCV 2022*, **13672** (2022), 511–528. https://doi.org/10.1007/978-3-031-19775-8_30

54. Y. Qin, C. Zhao, X. Zhu, Z. Wang, Z. Yu, T. Fu, et al., Learning meta model for zero-and few-shot face anti-spoofing, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **34** (2020), 11916–11923. https://doi.org/10.1609/aaai.v34i07.6866

55. H. Huang, D. Sun, Y. Liu, W. Chu, T. Xiao, J. Yuan, et al., Adaptive transformers for robust few-shot cross-domain face anti-spoofing, in *Computer Vision-ECCV 2022*, **13673** (2022), 37–54. https://doi.org/10.1007/978-3-031-19778-9_3

56. L. Li, X. Feng, Z. Xia, X. Jiang, A. Hadid, Face spoofing detection with local binary pattern network, *J. Visual Commun. Image Representation*, **54** (2018), 182–192. https://doi.org/10.1016/j.jvcir.2018.05.009

57. A. Roohi, S. Angizi, Efficient targeted bit-flip attack against the local binary pattern network, in *2022 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, (2022), 89–92. https://doi.org/10.1109/HOST54066.2022.9839959

58. N. Bousnina, L. Zheng, M. Mikram, S. Ghouzali, K. Minaoui, Unraveling robustness of deep face anti-spoofing models against pixel attacks, *Multimedia Tools Appl.*, **80** (2021), 7229–7246. https://doi.org/10.1007/s11042-020-10041-1

59. D. Deb, X. Liu, A. K. Jain, Unified detection of digital and physical face attacks, in *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, (2023) 1–8. https://doi.org/10.1109/FG57933.2023.10042500

60. T. K. Ho, The random subspace method for constructing decision forests, *IEEE Trans. Pattern Anal. Mach. Intell.*, **20** (1998), 832–844. https://doi.org/10.1109/34.709601

61. L. Breiman, Random forest, *Mach. Learn.*, **45** (2001), 5–32. https://doi.org/10.1023/A:1010933404324

62. F. T. Liu, M. T. Kai, Y. Yu, Z. Zhou, Spectrum of variable-random trees, *J. Artif. Intell. Res.*, **32** (2008), 355–384.

63. T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, preprint, arXiv:1603.02754.

64. T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Trans. Pattern Anal. Mach. Intell.*, **24** (2002), 971–987. https://doi.org/10.1109/TPAMI.2002.1017623

65. Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, S. Z. Li, A face antispoofing database with diverse attacks, in *2012 5th IAPR International Conference on Biometrics (ICB)*, (2012), 26–31. https://doi.org/10.1109/ICB.2012.6199754

66. P. Viola, M. Jones, Rapid object detection using a casacde of simple features, in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (2001). https://doi.org/10.1109/CVPR.2001.990517

67. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, preprint, arXiv:1409.1556.

68. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 770–778, . https://doi.org/10.1109/CVPR.2016.90

69. I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, preprint, arXiv:1412.6572.

70. Z. Sun, L. Sun, Q. Li, Investigation in spatial-temporal domain for face spoof detection, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2018), 1538–1542. https://doi.org/10.1109/ICASSP.2018.8461942

71. T. M. Oshiro, P. S. Perez, J. A. Baranauskas, How many trees in a random forest? in *Machine Learning and Data Mining in Pattern Recognition*, **7376** (2012), 154–168. https://doi.org/10.1007/978-3-642-31537-4_13

72. J. Lu, V. E. Liong, J. Zhou, Cost-sensitive local binary feature learning for facial age estimation, *IEEE Trans. Image Process.*, **24** (2015), 5356–5368. https://doi.org/10.1109/TIP.2015.2481327

73. J. Lu, V. E. Liong, J. Zhou, Deep hashing for scalable image search, *IEEE Trans. Image Process.*, **26** (2017), 2352–2367. https://doi.org/10.1109/TIP.2017.2678163

74. J. Lu, V. E. Liong, J. Zhou, Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, **40** (2018), 1979–1993. https://doi.org/10.1109/TPAMI.2017.2737538

75. Y. Duan, J. Lu, J. Feng, J. Zhou, Context-aware local binary feature learning for face recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, **40** (2018), 1139–1153. https://doi.org/10.1109/TPAMI.2017.2710183

AIMS Press