Hindawi
Security and Communication Networks
Volume 2021, Article ID 6652727, 16 pages
https://doi.org/10.1155/2021/6652727

Research Article

Towards Face Presentation Attack Detection Based on Residual Color Texture Representation

Yuting Du,[1] **Tong Qiao**,[1,2] **Ming Xu**,[1] **and Ning Zheng**[1]

[1]*School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China*
[2]*Zhengzhou Science and Technology Institute, Zhengzhou 450001, China*

Correspondence should be addressed to Ming Xu; mxu@hdu.edu.cn

Received 23 December 2020; Revised 7 February 2021; Accepted 27 February 2021; Published 16 March 2021

Academic Editor: Beijing Chen

Most existing face authentication systems have limitations when facing the challenge raised by presentation attacks, which probably leads to some dangerous activities when using facial unlocking for smart device, facial access to control system, and face scan payment. Accordingly, as a security guarantee to prevent the face authentication from being attacked, the study of face presentation attack detection is developed in this community. In this work, a face presentation attack detector is designed based on residual color texture representation (RCTR). Existing methods lack of effective data preprocessing, and we propose to adopt DW-filter for obtaining residual image, which can effectively improve the detection efficiency. Subsequently, powerful CM texture descriptor is introduced, which performs better than widely used descriptors such as LBP or LPQ. Additionally, representative texture features are extracted from not only RGB space but also more discriminative color spaces such as HSV, YCbCr, and CIE 1976 L∗a∗b (LAB). Meanwhile, the RCTR is fed into the well-designed classifier. Specifically, we compare and analyze the performance of advanced classifiers, among which an ensemble classifier based on a probabilistic voting decision is our optimal choice. Extensive experimental results empirically verify the proposed face presentation attack detector's superior performance both in the cases of intradataset and interdataset (mismatched training-testing samples) evaluation.

1. Introduction

Face authentication technology is widely deployed in real life. However, most existing face authentication systems are vulnerable to presentation attacks (PAs). For clarity, the bona fide and the PA samples are illustrated in Figure 1. Generally speaking, compared with the bona fide faces, the PA samples are generated by presenting spoofing artifacts toward face authentication system.

Since deep learning (DL) shows its outstanding potential in resolving image classification tasks, numerous DL-based methods are proposed by utilizing deep networks to extract deep features from images such as [1–6]. It is known that DL-based methods can achieve excellent performance when obtaining enough training data, but in face presentation attack detection task, the diversity and amount of training data is often not satisfied, and overfitting is also a vexing problem. To enable a presentation attack detection system be applicable to various environment, domain adaptation [7] manner is explored to resolve the overfitting. Moreover, similar to the two-stream strategy utilized in copy-move forgery [8], there is also two-stream-based method for learning fusion features to resolve PA detection problem [9].

Compared with DL-based methods, hand-crafted feature-based methods pay more attention to extract predefined specific patterns, which are more explainable. We can mainly divide these techniques into three categories: motion-related cue [10–13], image quality [14–16], and texture-based analysis [17–24]. Motion-related cue-based methods are highly robust in some specific cases, but the generalization ability is not satisfactory. Image quality artifact-based methods are not robust enough and computationally complex. By contrary, the performances of texture-based analysis methods are more preferable.

It is known that, in image forensics field, effective data preprocessing can obviously improve the algorithm's performance. For example, in [25], a Laplacian filter is used for input enhancement. And, in [26], the Schmid filter is used to

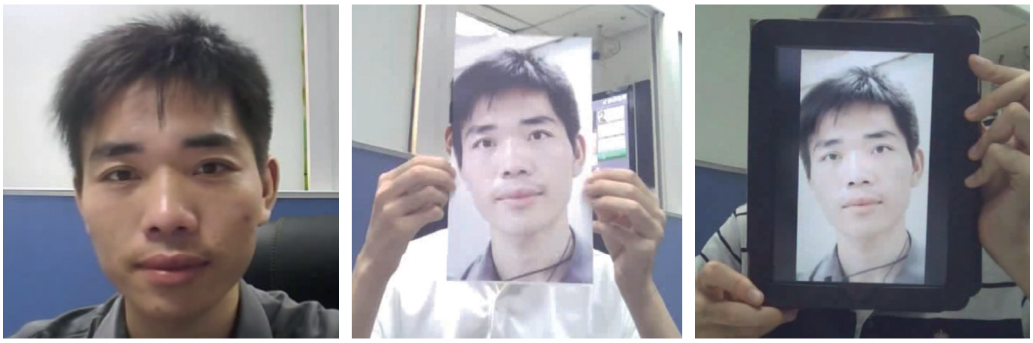


Figure 1: Cropped example face images extracted from the FASD. From the left to the right: genuine face, print attack, and replay attack, respectively.

enhance texture information. However, to the best of our best knowledge, in face antispoofing field, there is still a lack of effective measure of preprocessing. In this work, a novel perspective is introduced that nuisance noise can interference extracting representative features from face images, and we introduce a wavelet-based filter to preprocess the original image, which can successfully make the model perform better. The assumption is inspired by that in the process of using image sensors such as CCD and CMOS to capture images; due to the influence of the sensor material properties, electronic components, and circuit structure, various noises will be introduced, such as Gaussian noise, salt and pepper noise, speckle noise, shot noise, and white noise. However, such noise does not seem to be helpful for face PA detection. Therefore, analytical experiments are conducted to investigate how the difference changed between the bona fide and the PA faces by using residual (noise-free) images instead of original images (see Tables 1 and 2). For more intuitive, discrete wavelet transform is applied to conduct a similarity-based analysis, which is specifically described in Figure 2. By applying a discrete wavelet filtering (DW-filtering), compared with the original image, the similarity between the bona fide face and the PA from residual image is the lowest, meaning that the features extracted from both bona fide face and PA from residual image can be more discriminative than the others. Besides, since the effectiveness of texture analysis in color spaces is verified in [21], which utilizes two local texture descriptors (CoALBP and LPQ) and one classifier such as SVM, an assumption can be further drawn that if a high efficient classifier such as ensemble one, together with more discriminative descriptors for color residual texture representation is adopted, the performance of the detector can be further improved. The contributions of this paper can be summarized as follows:

Table 1: The Chi-square distances (i.e., d_{χ^2}) for different color channels in original images. Larger d_{χ^2} value is indicated in bold compared to Table 2.

Color channel	FASD	RAD	MSU
RGB-R	154.0	115.1	94.5
RGB-G	278.3	120.7	103.6
RGB-B	323.3	130.3	**114.1**
HSV-H	1062.7	766.0	717.0
HSV-S	404.4	**242.4**	304.7
HSV-V	188.1	115.4	**100.8**
YCbCr-Y	**253.6**	198.2	**103.8**
YCbCr-Cb	191.6	311.4	141.3
YCbCr-Cr	147.5	206.3	127.5
LAB-L	235.6	**120.1**	102.3
LAB-A	151.2	177.7	146.8
LAB-B	182.7	212.3	151.8

Table 2: The Chi-square distances (i.e., d_{χ^2}) for different color channels in residual images. Larger d_{χ^2} value is indicated in bold compared to Table 1.

Color channel	FASD	RAD	MSU
RGB-R	**165.0**	**118.6**	**99.3**
RGB-G	**289.4**	**122.9**	106.6
RGB-B	**328.9**	**130.6**	**114.4**
HSV-H	**1123.4**	**942.6**	**818.7**
HSV-S	**482.2**	239.7	**307.2**
HSV-V	**203.7**	**126.7**	99.4
YCbCr-Y	253.1	**199.3**	103.6
YCbCr-Cb	**200.7**	**313.0**	**253.4**
YCbCr-Cr	**246.9**	**212.8**	**250.1**
LAB-L	**239.6**	120.0	**107.4**
LAB-A	**418.7**	**325.3**	**287.5**
LAB-B	**198.7**	**213.8**	**262.6**

In RGB space, luminance and chrominance information cannot be effectively characterized. However, the concerning color information stored in different channels is of importance for generating more discriminative color features. Therefore, many works consider extracting features by using HSV, YCbCr space, or fusion of them. Nevertheless, for the differentiability of various color channels and the best combination of them, there is still a lack of deep exploration. In the following sections, we have conducted extensive analytical experiments and in-depth discussions on this issue. A total of four color spaces are taken into account, namely, RGB, HSV, YCbCr, and LAB.

Existing methods lack of effective data preprocessing. In fact, an effective preprocessing operation can significantly improve the performance of the detector. In the preprocessing stage of this work, we propose to adopt DW-filter for obtaining residual image, which effectively

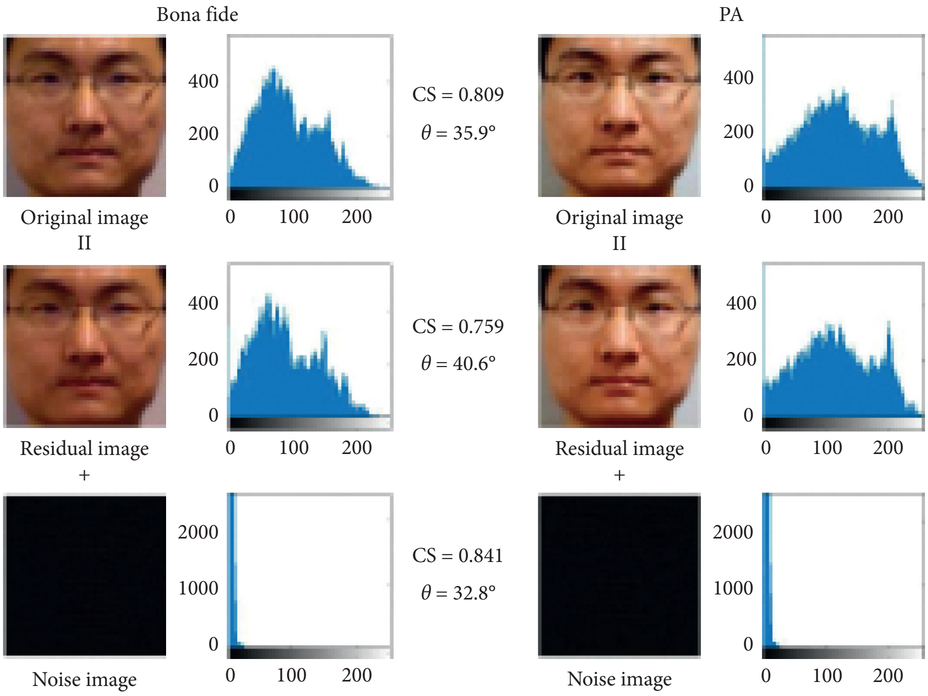


FIGURE 2: The CS of the bona fide and the PA samples. Residual image is obtained by DW-filter, where "Residual Image = Original Image – Noise Image." θ is the angle corresponding to the CS, which is inversely proportional to the image similarity.

alleviates the interference caused by nuisance noise while retaining valuable information for presentation attack detection. Meanwhile, extensive analytical experiments are conducted to further verify the effectiveness of the utilization of the residual image.

Among texture-based arts, the optimal choice of the descriptor is not well investigated. Thus, we mainly describe and analyze five widely used texture descriptors, namely, the CM, LBP, CoALBP, LPQ, and BSIF. According to the experimental results, the CM feature outperforms others in color spaces. Accordingly, our proposed RCTR is constructed relying on the powerful CM feature extracted in color channels of the residual image.

Most existing hand-crafted feature-based methods use single classifier such as SVM, which cannot always perform well. In this work, the performance of three widely adopted classifiers is well investigated, including LDA, SVM, and XGBoost. And, an ensemble classifier based on the probabilistic voting decision is designed. In the case of inter- or intradataset testing, our RCTR-based detector that employs the ensemble classifier shows satisfactory performance.

The remainder of this paper is organized as follows. In Section 2, the related works are presented. In Section 3, our proposed approach is described in detail. Three benchmark face presentation attack datasets are introduced in Section 4. In Section 5, we provide comprehensive experimental results and analysis. Last but not least, concluding remarks are drawn in Section 6.

2. Related Works

To address the challenge introduced by face presentation attacks, many presentation attack detection techniques have been proposed, which can be arbitrarily formulated into two categories: deep learning-based methods and hand-crafted feature-based methods. The specific overview is extended as follows.

2.1. Deep Learning-Based Methods. Deep learning can achieve promising results in the field of computer vision, which is also very effective when tackling face presentation attack detection task. In [2], CNN is utilized to extract deep features, and SVM is employed instead of fully connected layers for classification. Atoum et al. [27] present a two-stream network architecture to learn patch-based and depth-based features, and the classification result is determined by the fusion scores of both two streams. Rather than merely extracting spatial feature, a 3D-CNN structure is proposed in [6] to exploit the spatial-temporal features, which can capture more visual cues that are indeed useful for face presentation attack detection task. Meanwhile, a domain generalization regularization approach is incorporated for further enhancing the model generalization ability. Previous deep learning-based face presentation attack detection approaches formulate the task as a binary classification problem. Liu et al. [28] emphasize the importance of auxiliary supervision. Specifically, a CNN-RNN architecture is proposed to utilize depth map information and rPPG (remote Photoplethysmography) signs, which can both exploit spoof patterns across spatial and temporal domains. In [29], an augmented dataset is collected in a specific image synthesis way, which can further improve the robustness of the model.

DL-based methods usually have superior classification accuracy when training and testing samples belong to similar scenes. However, due to heavily relying on a large-scale well-designed dataset, the performance of many DL-based methods will sharply decrease when dealing with mismatched training and testing samples. Poor generalizability is more serious in earlier DL-based methods [3]. And, in recent works [30–32], such defect is significantly improved.

2.2. Hand-Crafted Feature-Based Methods. The methodologies in this category mainly rely on defining specific patterns in advance for extracting discriminative features. Given that face presentation attack samples tend to be static, motion analysis-based schemes are developed, such as eye blinking [10], mouth movement [11], and just holistic face region movement analysis [13]. In general, the biometric information can be successfully obtained by analyzing the optical flow in specific areas of the image. Although the motion-related cue-based methods perform well when dealing with print attack, they may fail to complete the task of replay attack detection, where the motion-related cue for presentation attack detection can be easily inferred. Besides, image quality also can be a vital measurement toward face presentation attack detection. Galbally et al. [15] propose to resolve presentation attacks by calculating prominent factors among 25 image quality metrics. Di et al. [16] introduce an image distortion analysis countermeasure by evaluating four presentation attack patterns: specular reflection caused by display device, image blurriness, chromatic distribution variation, and poor color diversity. However, due to heavy computation, these methods are not efficient enough. It is worth mentioning that although various hand-crafted feature-based methods are proposed, there is still a lack of effective preprocessing to further improve the performance of the detector.

In addition, the effectiveness of texture descriptors in resolving face presentation attack problems has been verified by some works. For instance, multiscale local binary pattern (MSLBP) descriptor is designed for face presentation attack detection in [17], and a novel facial texture representation is introduced by using the spatial and temporal extensions of the local binary pattern (LBP-TOP) [33]. Besides, it is worth noting that Boulkenafet et al. [21] present a novel and appealing face presentation attack countermeasure by using color texture features, based on the assumption that gray-scale images are often used to display illuminance information, while more helpful color information are discarded. In fact, the RGB image cannot completely separate the luminance and chrominance signals while color texture features can be well extracted from HSV and YCbCr spaces. It is well-known that print attacks utilize photos of legitimate users to fool the face recognition system, while replay attacks often utilize electronic device such as mobile or tablet. Due to the restriction of the limited color gamut, the fake faces presented on the display device often show color degradation.

The effectiveness of texture descriptors and color space features in resolving face presentation attack detection task are verified. However, the discriminative features are generally extracted from original pixels in spatial domain, which are more or less impacted by nuisance noise introduced during image capturing. Besides, the study of combining various texture features within different color spaces to achieve the optimal color texture features still remains open in this community. Additionally, to the best of our knowledge, one single classifier cannot always bring optimal prediction results, compared with the powerful ensemble classifier. In virtue of our theoretical and empirical analysis in this paper, those negative factors can lead to bad detection results when training samples are mismatched with testing samples. To address those challenges, dependent of residual image via DW-filtering, it is proposed to design a high efficient ensemble face presentation attack detector based on RCTR.

3. Proposed Method

In this section, we specifically present the RCTR-based face presentation attack detection method. For clarity, let us first illustrate the overall framework in Figure 3. First of all, face alignment is applied to calibrate the face region from full frame. Next, a DW-filter is utilized to process the high-frequency coefficients in order to obtain more discriminative residual image. Then, the residual image is transformed from RGB into another color space (e.g., YCbCr). Subsequently, texture descriptor is applied to extract rich texture information, in which a comprehensive representation is constructed by combining optimal descriptor feature vectors, namely, RCTR. Finally, we design an ensemble classifier with the effective strategy of probabilistic voting decision, which can successfully complete the task of face presentation attack detection.

3.1. Analysis of Color Space. The samples of PA face are passed through different cameras or printing mediums (such as photos, mobiles, and tablets), so they can actually be called a kind of recaptured image. Therefore, we can assume that when generating PA samples, inherent differences in color channel between the bona fide and PA images are introduced during the recapturing process. This is due to the color gamut caused by the display medium and other defects in color reproduction, such as display imperfection, or noise signals. Compared to bona fide face samples, the camera used to capture the target face photos also brings about imperfect color reproduction. Thus, it is reasonable to use color images instead of gray-scale images for face presentation attack analysis. RGB is widely used, but other color spaces are also worthy of attention. Because color component and luminance component cannot be perfectly characterized in RGB space, it can be better discriminated in other space such as HSV. There are various color spaces

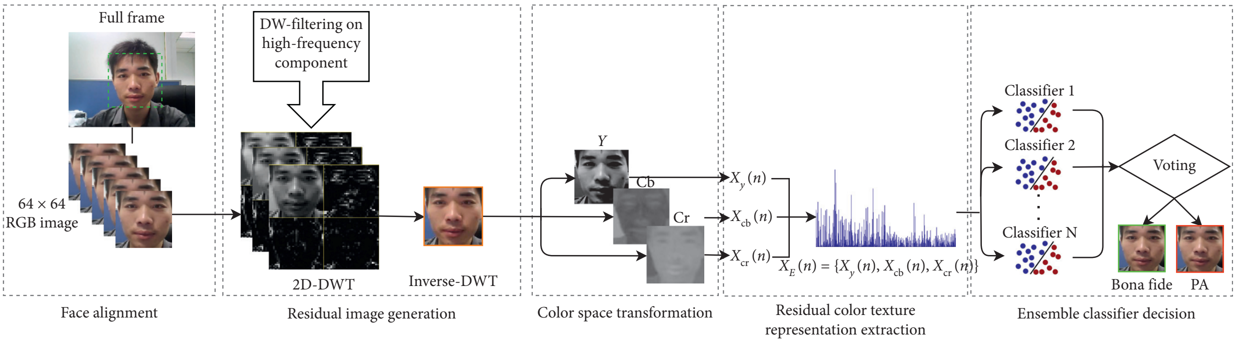

Full frame
DW-filtering on high-frequency component
64 × 64 RGB image
2D-DWT
Inverse-DWT
Y
Cb
Cr
$X_y(n)$
$X_{cb}(n)$
$X_{cr}(n)$
$X_E(n) = \{X_y(n), X_{cb}(n), X_{cr}(n)\}$
Classifier 1
Classifier 2
Classifier N
Voting
Bona fide
PA
Face alignment
Residual image generation
Color space transformation
Residual color texture representation extraction
Ensemble classifier decision

Figure 3: A pipeline of our proposed face presentation attack detection method, and YCbCr space is used here as an instance.

which have been proposed, and we consider analyzing bona fide face and PA images in four different color spaces: RGB, HSV, YCbCr, and LAB. Therefore, a metric is designed to examine which color space or channel is more distinguishable and details of the metric are as follows.

Firstly, for given image I with the size $l \times l$, the correlation coefficient between the adjacent pixels in each color component I^c ($c \in \{R, G, B_1, H, S, V, Y, \mathrm{Cb}, \mathrm{Cr}, L, A, B_2\}$) are calculated, which can be formulated as

$$m_i^c = \frac{\sum_{j=0}^{l-1}\sum_{k=0}^{l-1}\left(I_{j,k}^c - \overline{I}^c\right)\left(I_{j+1,k+1}^c - \overline{I}^c\right)}{\sqrt{\sum_{j=0}^{l-1}\sum_{k=0}^{l-1}\left(I_{j,k}^c - \overline{I}^c\right)^2\sum_{j=0}^{l-1}\sum_{k=0}^{l-1}\left(I_{j+1,k+1}^c - \overline{I}^c\right)^2}}, \tag{1}$$

where $\overline{I}^c$ represents the mean pixel value of I^c. For simplicity, we only consider the diagonally adjacent pixels. It can be drawn that the larger the m_i^c, the higher the relevance between the adjacent pixel values of given $\overline{I}^c$.

Subsequently, for given bona fide face image set, m_i^c of each image is calculated and the corresponding histogram H_{bf}^c can be constructed. And, for the given PA image set, the histogram H_{pa}^c can be obtained in the same way. Then, Chi-square distance is used to measure the similarity between the two histograms, which can be formulated as

$$d_{\chi^2}\left(H_{bf}^c, H_{pa}^c\right) = \sum_b \frac{\left(H_{bf}^c(b) - H_{pa}^c(b)\right)^2}{H_{bf}^c(b) + H_{pa}^c(b)}, \tag{2}$$

where b is the bin index of the histogram. Similarly, the larger the d_{χ^2}, more significant the difference between the bona fide images and the PA images.

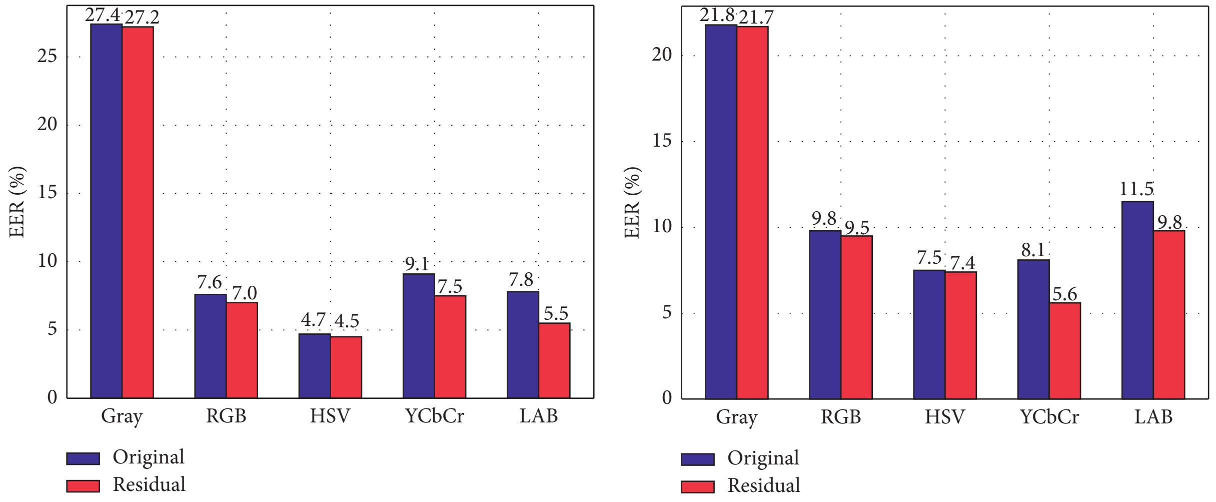
To evaluate the disparities between the bona fide face and the PA face in each color component, 10000 bona fide face images and 10000 PA face images are extracted from FASD, RAD, and MSU dataset, respectively, to perform analytical experiments. As introduced above, $m_i^c s$ of all images is calculated, the corresponding histograms H_{bf}^c and H_{pa}^c are obtained, and their $d_{\chi^2} s$ are also calculated, which can be seen in Table 1. Throughout the results of the three datasets, the d_{χ^2} values in RGB space are relatively stable (the maximum is 323.3, and the minimum is 94.5); this is because color components and luminance components are not well separated. As for the results on FASD, it can be observed that when using H channel, the d_{χ^2} value is 1062.7, which is significantly larger than any other channel. And, the result of the S channel is 404.4, which is the second largest. As for the V channel, the d_{χ^2} value is relatively small. This is meaning that the bona fide faces and the PA images are more distinguishable in color components (i.e., H and S channel) than in luminance component (i.e., V channel). As for YCbCr and LAB spaces, the differences between color component and luminance component are not as obvious as in HSV space. Similar conclusions can also be drawn from the results of RAD and MSU dataset.

Besides, only conducting analytical experiments are not enough to predict the actual situation; thus, extensive experiments are conducted to further investigate the benefit of color spaces transforming for face presentation attack detection (see Figure 4, for details).

3.2. Generation of the Residual Image.

3.2. Generation of the Residual Image. Face presentation attacks are implemented by printing human faces on various display media, such as A4 paper, mobile, and tablet screen. Though bona fide or PA samples are presented toward face authentication system, the nuisance noise is unavoidably introduced during image capturing process. A reasonable assumption can be made that nuisance noise existing in the face image, including bona fide and PA samples, might more or less impact the effectiveness of presentation attack detection, while the features extracted from the residual face image are more discriminative than that of original face image. Therefore, we propose to apply DW-filter for residual image extraction. It is important to study whether applying DW-filtering preprocessing operation in our scheme is effective to suppress nuisance noise from face image and meanwhile helpful to learn color texture features for presentation attack detection. To visually verify our hypothesis, we conduct the face image similarity-based analysis (see Figure 2 for illustration). By applying DW-filter, we segment the original face image to residual and noise one. Meanwhile, the statistical histogram of the pixels of each image is used to evaluate the similarity between two classes of face images, which is measured by the CS (cosine similarity):

$$\mathrm{CS}(X, Y) = \frac{X \cdot Y}{\|X\|\|Y\|} = \frac{\sum_{i=0}^{n=255} x_i \times y_i}{\sqrt{\sum_{i=0}^{n=255}\left(x_i\right)^2} \times \sqrt{\sum_{i=0}^{n=255}\left(y_i\right)^2}}, \tag{3}$$



(a)

(b)

Figure 4: The EER results of the CM feature extracted in various color spaces from both original images and residual images. (a) FASD. (b) MSU.

where $\|\cdot\|$ denotes the 2-norm and x_i or y_i represents the frequency in ith gray level of histogram from compared images. In Figure 2, we can observe that the CS between the noise images of bona fide and PA faces is 0.841. Meanwhile, we also observe that the CS between original images is 0.809, larger than 0.759 from residual images. That is because the noise components in face images are filtered out, which makes the inherent defects introduced by presentation attack operation to be more discriminative. In addition, we can also notice that the CS of noise image is higher than that of the original images, which further proves the interference effect of nuisance noise.

To further verify the effectiveness of the use of residual image, similar analytic experiments following the settings in Section 3.1 are conducted; the only difference is that the residual image is used instead of the original image (see Table 2, for illustration). It can be observed that compared to Table 1, most d_{χ^2} values for residual images are generally larger than that for original images; only a few color channels show a slight decrease (all larger d_{χ^2} values are indicated in bold in the table). Specifically, when using original images on RAD, the d_{χ^2} value of H channel is 766.0, and this value is increased to 942.6 when using residual images. Furthermore, for residual images, color components become more distinguishable in YCbCr and LAB spaces. Specifically, the d_{χ^2} value of Cb channel for residual images is 246.9, while the counterpart for original images is 147.5. And, the d_{χ^2} value of A channel for residual images is 418.7, while the counterpart for original images is just 151.2.

Based on the above analysis, we can draw that the discrimination between the bona fide and the PA faces can be further enhanced by adopting residual image instead of the original one. That is undoubtedly beneficial for presentation attack detection. Thus, prior to feature extraction such as residual color texture representation in this paper, it can hold true that we first proceed the preprocessing by using an effective filter.

The proposed algorithm needs to preprocess an inquiry face image by filtering. DW-filter serves as a useful tool to preliminary acquire the residual image (see Figure 2 for instance). DW-filter has performed its powerful advantage at decomposing high and low frequencies [34]. The application of 2D-DWT in image processing is mainly to decompose the inquiry image through multiscale decomposition. A 2D-DWT process over an original image I with the size $l\times l$ can be formulated by

$$f_{2D\text{-DWT}}(I)=\begin{bmatrix} I_{\mathrm{LL}} & I_{\mathrm{HL}} \\ I_{\mathrm{LH}} & I_{\mathrm{HH}} \end{bmatrix}, \tag{4}$$

where the original image I is decomposed into four subimages: I_{LL}, I_{HL}, I_{LH}, and I_{HH} with the size $l/2\times l/2$. I_{LL} corresponds to the approximation component (low frequency) of the image, while the remaining three I_{HL}, I_{LH}, and I_{HH} correspond to the horizontal detail component, vertical detail component, and diagonal detail component, respectively. As shown in Figure 2, when performing DWT filtering, the similarity between genuine face and fake face is reduced. In this case, the noise component is weakened after filtering, while the valuable information for presentation attack detection is preserved.

In particular, let us conduct DW-filtering proposed in [35], which can be formulated by

$$W_\lambda=\begin{cases}\operatorname{sgn}(|w|-\lambda), & |w|\geq\lambda,\\ 0, & |w|<\lambda,\end{cases} \tag{5}$$

where W represents the wavelet coefficients to be filtered, $\operatorname{sgn}(\cdot)$ is the sign function, and λ is the given threshold. In this work, we take the thresholding as a filter to preprocess face images. For instance, the sqtwolog threshold can be calculated by

$$\lambda=2\sqrt{2\log(l)}. \tag{6}$$

Specifically, let us introduce the process of the DW-filtering based on 2 layer decomposition in three steps:

The widely adopted haar wavelet base is selected, and the given original face image is decomposed by applying MALLAT decomposition algorithm [34]. Accordingly, the wavelet coefficients of each layer are successfully obtained.

Based on the given threshold λ, the high-frequency components obtained by decomposing each layer are quantized, while the low-frequency component remains unchanged.

By means of MALLAT reconstruction algorithm, the low-frequency component of the 2nd layer after decomposition and the high-frequency components of each layer are reconstructed by inverse DWT, and finally, the residual face image by wavelet thresholding is generated.

3.3. Feature Extraction by Texture Descriptor. Based on the previous analysis, we decide to extract texture features from multiple color channels in residual images. It should be noted that color texture features are obtained by applying descriptors not only in gray-scale image but also in color channels. That is because the color image can provide more valuable information for presentation attack detection, which is beneficial to improve detector's robustness and accuracy. In this work, the co-occurrence matrix [36] is employed, which is widely used in image texture analysis. Moreover, widely adopted descriptors such as the LBP [37], LPQ [38], CoALBP [39], and BSIF [40] are also introduced. In this section, we mainly overview these descriptors.

3.3.1. CM. The co-occurrence matrix (CM) describes the distribution of intensity and information about the relative position of adjacent pixels in the image, which can measure the correlation among adjacent pixels and hence gather valuable information from recurrent micropatterns. Before calculating CM, for given image I, first-order differential operator is applied to suppress the image content, namely,

$$\widehat{I}(x,y) = I(x,y) - I(x,y+1), \tag{7}$$

where (x,y) denotes the pixel coordinate and $\widehat{I}$ is the resulting image. It should be noted that only horizontal difference is considered here. As a result, the dynamic range of the image content is much narrower so that more reliable statistical description can be carried out. Subsequently, a truncating operation is conducted because there are too many distinct element values in the original image, which could result in huge dimension of the CM feature vector. The truncated image is calculated as follows:

$$T(x,y) = \begin{cases} \gamma, & \widehat{I}(x,y) \geq \gamma, \\ \widehat{I}(x,y), & -\gamma < \widehat{I}(x,y) < \gamma, \\ -\gamma, & \widehat{I}(x,y) \leq -\gamma, \end{cases} \tag{8}$$

where $\gamma > 0$ is the truncation threshold, and the result T is then used to compute the CM. Typically, a d order CM of the 2D array T can be obtained by

$$\begin{aligned} \mathrm{CM}(\theta_1,\theta_2,\ldots,\theta_d) = \frac{1}{N}\sum 1[T(x,y) = \theta_1, T(x+\Delta x, y+\Delta y) = \theta_2, \ldots, \\ T(x+(d-1)\Delta x, y+(d-1)\Delta y) = \theta_d], \end{aligned} \tag{9}$$

where $\theta_1,\theta_2,\ldots,\theta_d$ are the index, $1(\cdot)$ is the indicator function, N is the normalization factor, and Δx and Δy are the offsets. The effectiveness of the CM is validated in steganography detection [36] and face recognition [41]. However, in face presentation attack detection field, the use of the CM is not well explored.

3.3.2. LBP. The Local Binary Patterns (LBP) perform very well when depicting image structure information such as edges. The LBP is obtained via comparing each central pixel to its neighborhood one in the block, where the LBP features are described as a binary sequence, which can be formulated by

$$\mathrm{LBP}_{P,r}(x_c,y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^P, s(k) = \begin{cases} 1, & \text{if } k \geq 0, \\ 0, & \text{otherwise}, \end{cases} \tag{10}$$

where g_c denotes the value at the central pixel coordinate (x_c,y_c), while g_p, $p \in \{0,1,2,\ldots,P-1\}$, represents the value of the neighboring pixel in the block, and r denotes the radius. For instance, when $r = 1$, P equals to 8. Then, the binary patterns are collected by statistical histograms to represent the image texture information. In general, high robustness toward luminance variation, rotation invariance, and low-computational complexity are the advantages of LBP descriptor. When a face image is tested, we cannot guarantee that it is correctly presented in front of a digital camera of presentation attack detector. Thus, the robustness of resisting rotation attack is crucial. However, the LBP feature contains only intensity relationships between adjacent pixels and lack of spatial relationship information, which raises the performance limitation.

3.3.3. CoALBP. For the sake of compensating the missing spatial relationship information in the LBP features, the co-occurrence of adjacent local binary patterns (CoALBP) is proposed in [39]. In this method, two simplified LBP configurations, denoted as LBP (+) and LBP (×), are introduced. LBP (+) considers two horizontal and two vertical pixels, while LBP(×) considers four diagonal pixels. Before calculating the co-occurrence information of LBPs, each LBP is transformed to its vector form by using Kronecker delta:

$$V_i(B) = \delta_{i,l(\mathrm{lbp}(B))},$$
$$\delta_{a,b} = \begin{cases} 1, & \text{if } a \neq b, \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where $i \in \{0, 1, 2, \ldots, n-1\}$, n is the number of neighbor pixels, B is the position vector in an image intensity I, and $l(\mathrm{lbp}(\cdot))$ denotes a decimal number label of $\mathrm{lbp}(\cdot)$. For example, if the given binary sequence is 0010, the corresponding label is 2. If all possible LBP label values are in the range $[0, N]$ $(N = 2^n)$, an $N \times N$ autocorrelation feature matrix H can be calculated by

$$H(D) = \sum_{B \in I} V(B)V(B+D)^T, \tag{12}$$

where D is the displacement vector between two LBPs. Four displacement vector are set as follows: $D_1 = (\Delta B, 0)^T$, $D_2 = (\Delta B, \Delta B)^T$, $D_3 = (0, \Delta B)^T$, and $D_4 = (-\Delta B, \Delta B)^T$, which correspond to the direction of $0^\circ, 45^\circ, 90^\circ$, and 135°. At last, the four resulting matrices are concatenated to form the final CoALBP feature. It should be noted that although the CoALBP descriptor preserves more spatial information than LBP, the high dimension of CoALBP feature increases the computation cost of training a classifier.

3.3.4. LPQ. The local phase quantization (LPQ) is originally proposed by [38] to solve the problem of inaccurate classification caused by image blurring. The LPQ descriptor uses local phase information, which is extracted through the short time Fourier transform (STFT) based on the square region. The resulting STFT within the region of $g \times g$ surrounding the central pixel position m from the given image is defined by

$$F_{\mathbf{u}}(m) = w_u^T \mathbf{x}, \tag{13}$$

where w_u represents the basis vector of the 2D discrete Fourier transform at the frequency u and $\mathbf{x}$ denotes the vector containing all pixels in the region of $l \times l$. Specifically, the Fourier complex coefficients are calculated at four 2D frequencies: $u_0 = (s, 0)^T$, $u_1 = (s, s)^T$, $u_2 = (0, s)^T$, and $u_3 = (s, s)^T$, where s is a small scalar and $s \ll 1$. Then, the basic LPQ feature can be formulated by

$$Q(m) = [\mathrm{RC}\{Q^c(m)\}, \mathrm{IC}\{Q^c(m)\}],$$
$$Q^c(m) = \{F_{u_0}(m), F_{u_1}(m), F_{u_2}(m), F_{u_3}(m)\}, \tag{14}$$

where RC{·} and IC{·} mean to return the real component and imaginary component of a complex number, respectively. In addition, each element of $Q(m)$ is quantized as a binary sequence by a preliminary defined function. At last, the resulting binary sequence is represented as decimal integer values in the range [0, 255] and collected into feature histogram, which is similar to LBP. While LPQ is known to possess invariance to blurring effects, as discussed in [16], it is possible that image blurring is relevant to face presentation attack.

3.3.5. BSIF. Without loss of generality, the optimal selection of local features can effectively capture the relevant structure characteristics of the image. Alternatively, the binarized statistical image features (BSIF) [40] are adopted in a manner, in which an inquiry image is convolved with a linear filter, and then, the binary code of the filter response is obtained. By means of independent component analysis (ICA), the weight values of the filters are learned from a set of natural image patches by maximizing the statistical independence of the filter responses. Given an image block C and a bank of linear filters with the same size, the convolutional response r_i is computed by

$$r_i = C * W_i, \tag{15}$$

where W_i denotes the filter, $i \in \{1, \ldots, n\}$. Specifically, in this work, 8 filters are used (i.e., $n = 8$). And, then, the binarized feature is obtained:

$$b_i = \begin{cases} 1, & \text{if } r_i \geq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

It should be noted that the filter W_i has been well-trained by learning a set of heterogenous natural images which is different from the face images. Therefore, the BSIF features can avoid tedious filter design and parameter tuning. Moreover, the BSIF descriptor is capable of serving as a general descriptor to deal with various presentation attack scenarios in the practical detection.

3.4. Design of the Classifier. After extracting valid features, an efficient and accurate classifier is supposed to design. Various classifiers are adopted in face presentation attack detection (see [12, 42–44], for instance). In general, the monotone classifier structure equipped with fixed parameters possibly leads to the deviation of classification results. In order to achieve high level detection accuracy and generalization ability, we intend to investigate the following classifiers and select the optimal scheme of designing a classifier based on the proposed color residual texture representation.

3.4.1. LDA. Linear discriminant analysis (LDA) is a supervised approach that is widely adopted in the field of face recognition [45] and face presentation attack detection [12], which can be used for both dimensionality reduction and classification. The objective of LDA is to find a proper projection that maximizes the between-class scatter matrix and minimizes the within-class scatter matrix in the projective feature space. In the past, the image data was directly used as input, but when dealing with the high-dimensional face data, LDA often suffers from the small sample size problem. In this work, we extract texture descriptors with strong expressiveness from face images and relatively low dimension features are extracted. Then, LDA can also be used as a classifier to be considered.

3.4.2. SVM. Support vector machine (SVM) is a kind of classifier of generalized model for binary classification tasks based on supervised learning. By utilizing the kernel method, nonlinear classification tasks can also be accomplished. Due to the outstanding property of sparsity and robustness, SVM is often used when resolving face recognition missions [46]. The decision boundary of SVM is the maximum margin hyperplane for the solution of learning samples. Furthermore, SVM uses hinge loss functions to calculate empirical risks and adds regularization terms to the solution system to optimize structural risks. Face presentation attack detection can be considered as a binary classification task, and support vector machines are classifiers with the potential to cope with such task. More importantly, the feature size obtained by our hand-crafted feature-based method is relatively large, and SVM performs well when learning high-dimensional feature vectors.

3.4.3. XGBoost. By optimizing the boosting algorithm on the basis of gradient boosting decision tree (GBDT), extreme gradient boosting (XGBoost) has been employed to resolve the classification and regression problems in many fields [47]. In fact, XGBoost is still based on the tree model. Hundreds of tree models with low classification accuracy are combined to iterate continuously, and each iteration generates a new tree. XGBoost adds a regular term to the cost function to control complexity. From the perspective of bias-variance trade off, the regular term reduces the variation of the model, makes the learned model simpler, and prevents overfitting. When conducting face presentation attack detection, a detector based on XGBoost classification possibly produces superior generalization ability dealing with heterogenous data.

3.4.4. Ensemble Classifier. As [48] states, to make an ensemble decision, constituent classifiers should be heterogenous, and meanwhile, their classification performances should be comparable. Accordingly, three base classifiers (LDA, SVM, and XGBoost) are selected in our well-designed ensemble classifier. Actually, we have also tried other kinds of classifiers, such as Naive Bayesian and Decision Tree. However, these two classifiers are not adopted in our design due to unsatisfying performance. The scheme of voting decision can be referred to as a soft voting, which is not a simple majority rule. Specifically, the average of the probability that all model prediction samples are in a certain class is taken as the threshold, and the corresponding class with the highest probability will bring the final prediction result. As Figure 5 illustrates, Classifier 1 and Classifier 2 both predict the test sample "Bona Fide," and only Classifier 3 outputs "PA," while after the soft voting decision, the final result is still "PA." The experimental results in Section 5.4 also can verify that our carefully designed voting scheme produces better performance than using single classifier.

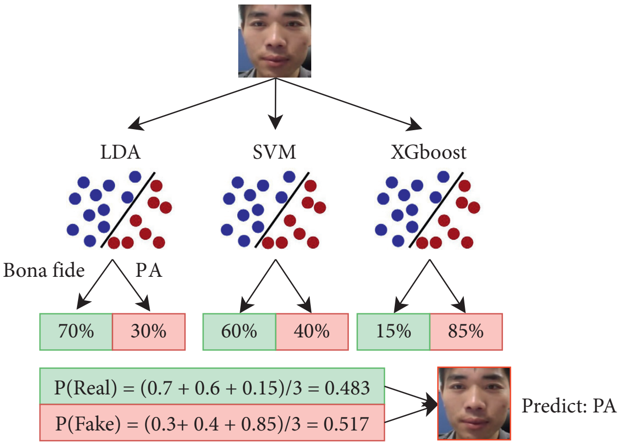

LDA
SVM
XGboost
Bona fide
PA
70%
30%
60%
40%
15%
85%
P(Real) = (0.7 + 0.6 + 0.15)/3 = 0.483
P(Fake) = (0.3+ 0.4 + 0.85)/3 = 0.517
Predict: PA

Figure 5: A toy example of the voting decision. For clarity, three base classifiers are used here.

4. Description of the Benchmark Datasets

In this work, four challenging benchmark datasets are used to evaluate our proposed detector: CASIA Face Antispoofing Dataset (FASD), Replay-Attack Dataset (RAD), MSU Mobile Face Spoof Dataset (MSU), and ROSE-YOUTU Face Liveness Detection Dataset (ROSE). For clarity, a summary of the four datasets is illustrated in Table 3. Detailed descriptions of the four datasets are given as follows.

4.1. CASIA Face Antispoofing Dataset. The CASIA Face Antispoofing Dataset [49], released in 2012, consists of 600 video clips from 50 different clients. There are three attack types involved. (1) *Warped Photo Attack*. The photograph of the legitimate client is presented to the camera, and the movement of the face is simulated by bending the photo. (2) *Cut Photo Attack*. The eye area in the face photo is cut out and a person blinks behind the paper hole. (3) *Replay Video Attack*. High-resolution video of face is displayed on a tablet. There are three imaging quality level used to record the whole real accesses and spoofing attacks. (1) *Low-quality*, with 640 × 480 resolution, captured by a cheap USB-camera. (2) *Normal-quality*, with 480 × 640 resolution, captured by another USB-camera better than the former. (3) *High-quality*, with 1280 × 720 resolution, captured by a Sony NEX-5 camera. The recordings of the total 50 clients are established, in which 20 clients are split into training set and remaining 30 clients into testing set.

4.2. Replay-Attack Dataset. The Idiap Replay-Attack Dataset [19], released in 2012, includes 1200 video recordings of both real accesses and spoofing attacks from 50 subjects. The video recordings are collected at two different stationary conditions. (1) *Controlled*. Uniform background scenes and lighting equipment are applied. (2) *Adverse*. Background is not uniform and only natural day-light illuminates. Under the same environments, each client is taken two high-resolution photos with Canon PowerShot SX150 IS and iPhone 3GS, respectively. These recordings are utilized to fabricate the spoofing attack samples. In total, there are three attack

Table 3: A summary of the four publicly available face spoofing datasets FASD, RAD, MSU, and ROSE.

Dataset	Release time	Subjects	Video number	Acquisition camera	Attack scenarios
FASD	2012	50	600 (150 genuine, 450 fake)	Low-quality camera (640 × 480) Normal-quality camera (480 × 640) Sony NEX-5 camera (1280 × 720)	(1) Warped photo (2) Cut photo (3) Replay video
RAD	2012	50	1200 (200 genuine, 1000 fake)	MacBook 13″ camera (320 × 240)	(1) Print (2) Mobile (3) High-def
MSU	2014	55 (35 available)	280 (110 genuine, 330 fake)	MacBook air 13″ camera (640 × 480) Google nexus 5 camera (720 × 480)	(1) Printed photo (2) Replayed video
ROSE	2018	20	3350 (500 genuine, 2850 fake)	Hasee phone (640 × 480) Huawei phone (640 × 480) iPad 4 (640 × 480) iPhone 5s (1280 × 720) ZTE phone (1280 × 720)	(1) Printed paper (2) Video replay (3) Masking

scenarios. (1) *Print*. High-resolution face photos are printed on A4 papers and displayed in front of the camera. (2) *Mobile*. High-resolution pictures and videos are displayed on an iPhone screen. (3) *High-def*. The photographs and videos are shown on an iPad screen with 1024 × 768 resolution. All recordings of 50 clients are partitioned into three disjoint subsets: (1) *Train*, (2) *Development*, and (3) *Test*, with 15, 15, and 20 clients, respectively.

4.3. MSU Mobile Face Spoof Dataset. The MSU Mobile Face Spoof Dataset [16], released in 2014, consists of 440 video clips of genuine and fake faces taken from 55 clients in total, while 280 recordings corresponding to 35 clients' subset are available. Two types of cameras are used to collect the data: a built-in camera of Macbook Air 13," referred to as laptop camera, with 640 × 480 resolution and a front-facing camera of Google Nexus 5, referred to as Android camera, with a resolution of 720 × 480. There are two spoofing attack types included. (1) *Printed Photo*. To generate the printed attack samples; a HD photograph of the client's face is captured by the Canon 550D camera, with 5184 × 3456 resolution. Then, the photo is printed on an A3 paper using a HP color printer. (2) *Video Replay*. The video of the client's face is first recorded using a Canon 550D camera and an iPhone 5S back-facing camera. The Canon camera is used to capture a HD video with 1920 × 1088 resolution, which is replayed on an iPad Air screen. And, the iPhone 5S is used to capture another HD video with 1920 × 1080 resolution, which is replayed on the iPhone 5S screen.

4.4. ROSE-YOUTU Face Liveness Detection Dataset. The ROSE-YOUTU Face Liveness Detection Dataset [7], released in 2018. ROSE dataset consists 3350 videos from 20 clients. For each client, there are 150–200 video clips with the average duration about 10 seconds. Five types of mobiles are used to collect the dataset: a Hasee smart-phone with the resolution of 640 × 480, a Huawei smart-phone with a resolution of 640 × 480, an iPad 4 with the resolution of 640 × 480, an iPhone 5s with resolution of 1280 × 720, and a ZTE smart-phone with resolution of 1280 × 7200. Three spoofing attack types are considered: (1) printed paper attack: to generate fake samples; still printed paper and quivering printed paper (A4 size) are used, (2) video replay attack: face videos are displayed on Lenovo LCD screen and Mac screen, and (3) masking attack: masks with and without cropping are presented.

5. Experimental Results and Analysis

5.1. Experimental Setup. As prior works [19, 21, 50], the face video recordings in FASD, RAD MSU, and ROSE datasets are split into single-face region frame, and frame-based experiments are conducted. All face images are normalized into 64 × 64 size after face alignment; the facial landmarks are localized by using Dlib 19.14.0 [51]. The parameter settings of the descriptors are shown as follows: when extracting the CM feature, two first-order differential operators are applied (in horizontal direction and vertical direction), the truncation threshold $\gamma = 2$, and the order is set as $d = 3$. And, the offsets are chosen as $(\Delta x, \Delta y) \in \{(0, 1), (1, 0)\}$. As for LBP feature, the parameters $P = 8$ and $R = 1$. As for CoALBP feature, LBP (+) is used with radius $R = 1$ and the corresponding $\Delta B = 2$. The parameters for the LPQ descriptor are $g = 7$ and $s = 1/7$. At last, the filter size of BSIF features is set as 7 × 7. The dimension of the texture feature extracted by using the CM, LBP, CoALBP, LPQ, and BSIF on single channel is 75, 59, 1024, 256, and 256, respectively. Additionally, scikit-learn toolkit [52] is used for model training and parameter fine-tuning.

In the following experiments, equal error rate (EER) is used as a metric. In general, a threshold is adopted to calculate the false reject rate (FRR) and the false accept rate (FAR). When these two rates are equal by adjusting the threshold, the common value is referred to as EER. Besides, HTER also serves as another metric for evaluation (advised on RAD), which can be formulated by

$$\mathrm{HTER} = \frac{\mathrm{FAR}(\tau, D) + \mathrm{FRR}(\tau, D)}{2}, \tag{17}$$

where τ is the value of the EER estimated on the dataset D. It should be noted that the smaller EER or HTER represents the better detection result.

5.2. Validation of the Residual Color Texture Representation. In this section, the CM descriptor is used as an instance to verify the effectiveness of employing RCTR. Both benchmark FASD and MSU are used for testing. In Figure 4, the EER of the CM features extracted from gray-scale image, RGB, HSV, YCbCr, and LAB spaces are presented, where the SVM classifier is used. As can be clearly observed, the results obtained by using residual images are generally better than that of using original images both on the two datasets. Thus, it can hold true that, by using the residual image instead of the original image, the interference of nuisance noise can be effectively reduced, while more discriminative features for presentation attack detection can be extracted. More importantly, the effectiveness of color space transforming can also be verified in Figure 4. When considering the EER of the CM features extracted from residual images, the worst result is shown in the case of gray scale both on FASD and MSU. Besides, the lowest EER on FASD is 4.5% when using HSV space, and the best performance on MSU is 5.6% in the case of YCbCr.

5.3. Performance Comparison of Different Texture Descriptors. In this part, the performance of the LBP, CoALBP, LPQ, BSIF, and CM descriptors are evaluated on FASD, where SVM classifier is employed, as shown in Figure 6. It can be observed that the EERs of the CM descriptor (brown column) is obviously lower than that of the other four types of descriptors in the cases of RGB, HSV YCbCr, and LAB, and the CoALBP descriptor (red column) performs best in the case of gray scale. Since the performance of all descriptors is relatively poor in gray-scale space, we only consider using RGB, HSV, YCbCr, and LAB spaces. Thus, the CM descriptor is selected to construct the final RCTR.

5.4. Evaluation of Different Classifiers. Subsequently, the EER results of the CM features on benchmark FASD by employing different classifiers are presented, as shown in Table 4. And, for fair comparison, the average EERs of each classifier is also presented. It can be observed that, basically, our proposed ensemble classifier maintains the lowest EER in most cases except in gray scale. Moreover, the average EER of ensemble classifier is 10.7%, which is still the lowest among four powerful classifiers. Obviously, our proposed probabilistic decision-based ensemble classifier can perform better than using single classifier such as LDA, SVM, or XGBoost.

5.5. Fusion of the Residual Color Texture Representation. In this section, the fusion performance of color spaces for RCTR is well-explored. A total of four color spaces are considered, namely, RGB, HSV, YCbCr, and LAB. As discussed above, the CM descriptor is selected to extract texture features from residual images to construct the RCTR, and the ensemble classifier is employed. Extensive experiments based on different color space fusions are conducted, in which the benchmark FASD and MSU are used for evaluation, as can be seen in Table 5. Furthermore, the performance of the combination of only color components is also explored. Specifically, {H,S,Cb,Cr} means the RCTR extracted from H, S, Cb, and Cr channels.

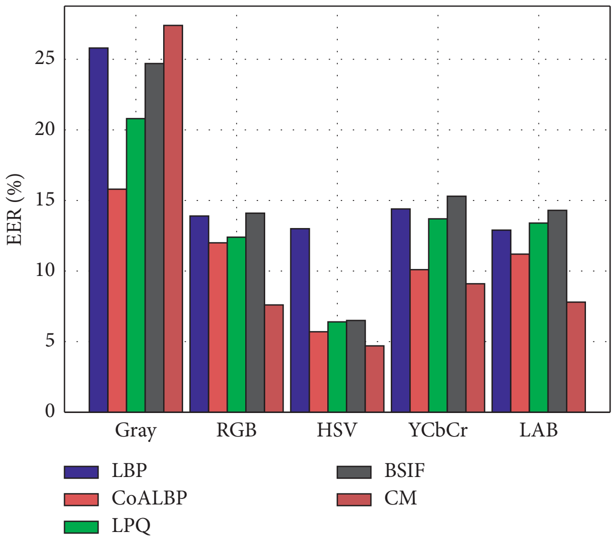


Figure 6: The EER results of the LBP, CoALBP, LPQ, BSIF, and CM features extracted from various color spaces.

Table 4: The EER results of the CM features extracted from various color spaces when using LDA, SVM XGBoost, and Ensemble Classifier.

Color space	LDA	SVM	XGBoost	Ensemble
Gray	30.6	27.4	**26.1**	27.2
RGB	8.3	7.6	6.9	**6.3**
HSV	5.5	4.7	5.6	**4.4**
YCbCr	9.7	9.1	9.0	**8.1**
LAB	8.5	7.8	9.7	**7.5**
Average	12.5	11.3	11.5	**10.7**

Table 5: The performance of various color space combinations of RCTR when employing ensemble classifier.

Color space fusion	FASD	MSU	Average
	EER	EER	EER
RGB + HSV	2.4	5.3	3.95
RGB + YCbCr	3.5	3.2	3.35
HSV + YCbCr	2.1	2.3	2.20
RGB + HSV + YCbCr	**1.6**	2.5	2.05
HSV + YCbCr + LAB	2.0	2.3	2.15
All spaces	1.8	**2.0**	**1.90**
{H,S,Cb,Cr}	4.9	5.5	5.20

As shown in Table 5, when combining the features of all four color spaces, the optimal performance of RCTR can be achieved on MSU (with the EER of 2.0%). As for FASD, when combining RGB, HSV, and YCbCr spaces, the lowest EER (1.6%) is obtained. Meanwhile, the EER of the RCTR extracted from {H,S,Cb,Cr} is 4.9% and 6.5%, respectively, which is not as good as combining all color spaces.

When considering the average value, the EER when combining all four spaces is the lowest (1.9%). And, it can be clearly observed that when combining three color or four spaces, the EERs of the detector are generally lower than those only combining two spaces. Then, we can draw that, in most cases, by combining the RCTR features of more color spaces, the performance of our face PA detector can be further improved.

5.6. Intradataset Performance in Comparison with the State of the Art. In this section, we evaluate the performance for identifying the bona fide and the PA images in the case that the training and testing data are matched. Tables 6–8 present the experimental results of our proposed method and the state-of-the-art techniques ([12, 16, 21, 33, 53, 54, 55, 56, 57] for hand-crafted feature-based methods and [2, 3, 9, 27, 58, 59] for DL-based methods). The results of the RCTR combining four color spaces are used for comparison. It should be noted that, as reported in [16], because only a portion samples (high-quality) of FASD were used for evaluation, for fair comparison, the result is not listed in Table 6. And, since the EER is not adopted in [16, 55], we only cite HTER results on RAD.

From Table 6, we can observe that DRL-FAS [59] outperforms other methods both on FASD and RAD. Our method outperforms most methods except [59] on FASD and shows competitive performance on RAD. Motion Mag algorithm [12] also achieves the best HTER on RAD, but suffers significant degradation when testing on FASD. Meanwhile, it can be seen that the performance of most hand-crafted feature-based approaches and DL-based methods are satisfactory on RAD. The reason lies on that when collecting the data of RAD; the photo capture condition is relatively simple, i.e., only one kind of camera is adopted.

As can be clearly observed in Table 7, our RCTR-based detector achieves the lowest EER on MSU (2.0%). And, from Table 8, it is observed that DRL-FAS [59] achieves best performance on ROSE, with an EER of 1.8%. Our method gets the second place, with an EER of 10.7%, which is the best performance among hand-crafted feature-based methods [21, 57] and better than DL-based method [60]. In conclusion, these results indicate that the bona fide and the PA images can be accurately identified by employing our proposed RCTR-based ensemble classifier.

Table 6: Performance comparison with the state-of-the-art methods on FASD and RAD. “-” represents that the results are not available.

Method	FASD	RAD	
	EER	EER	HTER
LBP-TOP [33]	10.0	7.9	7.6
LDP-TOP [53]	8.9	2.5	1.8
Motion Mag [12]	14.4	0.2	**0.0**
IDA [16]	—	7.4	—
Dynamic [54]	21.8	5.3	3.8
Spectral cubes [55]	14.0	—	2.8
CVLBC [56]	6.5	1.7	0.8
Color LBP [57]	7.1	0.9	4.9
Color [21]	2.1	0.4	2.8
Deep CNN [3]	7.4	6.1	2.1
Partial CNN [2]	4.5	2.9	4.3
LBP-Net [58]	2.5	0.6	1.3
Fusion CNN [27]	2.7	0.8	0.7
MobileNet + attention [9]	4.2	0.1	0.3
ResNet + attention [9]	3.1	0.2	0.4
DRL-FAS [59]	**0.2**	**0.0**	**0.0**
RCTR-all spaces (ours)	1.8	0.7	2.1

Table 7: Performance comparison with the state-of-the-art methods on MSU.

Method	EER
LBP + SVM baseline	14.7
DoG-LBP + SVM baseline	23.1
IDA [16]	8.5
LDP-TOP [53]	6.5
Color LBP [57]	10.6
Color [21]	4.9
RCTR-all spaces (ours)	**2.0**

Table 8: Performance comparison with the state-of-the-art methods on ROSE.

Method	EER
LBP + SVM baseline	34.1
Color LBP [57]	27.6
Color [21]	13.9
De-spoofing [60]	12.3
DRL-FAS [59]	**1.8**
RCTR-all spaces (ours)	10.7

Table 9: Interdataset testing comparison on the FASD dataset versus the RAD in terms of HTER.

Method	Train FASD Test RAD	Train RAD Test FASD	Average
LBP-TOP [33]	49.7	60.6	55.2
Motion Mag [12]	50.1	49.7	49.9
Spectral cubes [55]	34.4	50.0	42.2
Deep CNN [3]	48.5	45.5	47.0
Auxiliary [28]	27.6	**28.4**	28.0
De-spoofing [60]	28.5	41.1	34.8
STASN [29]	31.5	30.9	31.2
MobileNet + attention [9]	30.0	33.4	31.7
ResNet + attention [9]	36.2	34.7	35.5
DRL-FAS [59]	28.4	33.2	30.8
SSR-FCN [61]	**19.9**	41.9	**27.0**
RCTR-all spaces (ours)	37.1	42.0	39.6
RCTR-{H,S,Cb,Cr} (ours)	31.8	39.5	35.7

5.7. Interdataset Performance Comparison with the State of the Art. To evaluate the performance of the detector when training and testing samples are mismatched, cross testing among all three datasets is conducted. The HTER results of our RCTR-based detector when combining all four spaces and only using H, S, Cb, and Cr channels are presented in Table 9. It is observed that SSR-FCN [61] performs best when training on FASD and testing on RAD (with an HTER of 19.9%), but in another case, the performance of SSR-FCN is relatively poor (41.9%). When training on RAD and testing on FASD, auxiliary [28] outperforms other methods (with the EER of 28.4%). As for our proposed method (RCTR-{H, S, Cb, Cr}), the HTER is 31.8% and 39.6%, respectively, which significantly outperforms the methods proposed in [3, 12, 33, 55] while comparable with outstanding arts in [9, 28, 29, 59, 60]. It is worth noting that the HTER of RCTR-all spaces is higher than RCTR-{H,S,Cb,Cr}. This phenomenon can be explained as follows: when capturing the face records, the scene's brightness condition of different datasets is not consistent, so the RCTR feature extracted in complete color spaces containing the luminance

Table 10: Interdataset testing comparison with color texture-based methods on FASD, RAD, and MSU datasets in terms of HTER.

Method	Training	FASD		RAD		MSU		Average
	Testing	RAD	MSU	FASD	MSU	FASD	RAD	
Color LBP [57]		47.0	36.6	39.6	35.2	49.6	42.0	41.7
Color [21]		**30.3**	20.4	**37.7**	34.1	46.0	**33.9**	33.7
RCTR-{H,S,Cb,Cr} (ours)		31.8	**19.1**	39.5	**29.0**	**41.3**	34.4	**32.5**

information is not as good as in the color component, i.e., H, S, Cb, and Cr.

Besides, it can be observed that when training on FASD and testing on RAD, the result is better than training on RAD and testing on FASD. The reason lies in FASD which has more types of cameras and more attack scenarios; thus, the detector is more robust. However, the manner of collecting the recordings of RAD dataset is relatively simple, and the lack of diversity in training data leads to poor performance of the detector when testing on new dataset.

In Table 10, more comprehensive experiments are conducted to compare our method with other color texture-based methods [21, 57]. It can be seen that when training on FASD and testing on MSU, the HTER of our proposed detector is lowest. Similarly, our proposed method performs best in half of the cases. In addition, the average HTER of our proposed detector is 32.5%, which is also the lowest. The well performance of our proposed algorithm using color residual texture representation when testing on mismatched samples can be attributed to the generalization ability of the CM feature and the highly robust ensemble classifier.

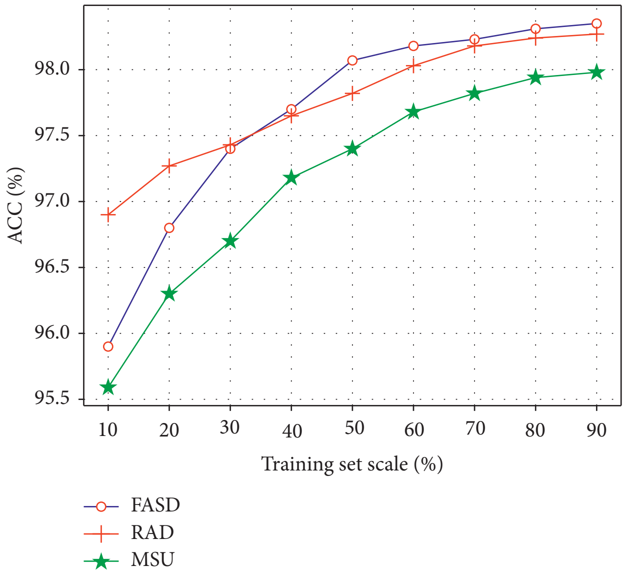


Figure 7: Performance of the proposed method versus the training set scale.

5.8. Performance versus Training Set Scale. In this part, we investigate on how the scale of training data impacts the performance of the proposed method. Specifically, the training set scale is increased from 10% to 90%, with a step of 10%, and the remaining data are used for validation. 10-folds' validation experiments are conducted, and each experiment randomly selects face images to form the training set; the average of the results are taken as the final result. Prediction accuracy (ACC for short) is used as metric, that is, the ratio of correct predictions to the total testing samples. As illustrated in Figure 7, as the scale of training data increases, the ACC of our proposed presentation attack detector is gradually improved. And, when using only 10% training data, the ACC of our RCTR-based detector on all three datasets is higher than 95.5%. The empirical study indicates that our proposed method can achieve excellent prediction accuracy with a small-scale training data. In addition, since DL-based methods are data-driven, so the performance of them is likely to be unsatisfactory when there is insufficient training data.

5.9. Time Complexity Analysis. We conduct time consumption statistical experiments to analyze the processing time. All methods considered are implemented by using Matlab2017a and Python 3.6 on an Intel Core i7 2.8 GHz CPU and 16 GB RAM PC. A total of 500 videos are used, and the number of frames of each video is between 300 and 400. The average processing time of each video is recorded, which is shown in Table 11. It can be observed that our method can achieve a competitive time consumption compared with other methods (with an average processing time of 15.6 second), which indicates the good real-time detection ability of the proposed method. Furthermore, our method has better detection accuracy compared with other methods.

Table 11: Processing time (per video) of our method and some baseline methods.

Method	Time (second)
LBP + SVM baseline	10.3
Color LBP [57]	12.2
Color [21]	21.9
RCTR-all spaces (ours)	15.6

6. Conclusion

In this paper, we propose a RCTR-based detector to address the challenge raised by face PA. First, by considering the nuisance noise existing in face image, a DW-filter is applied to eliminate such interference, after which more discriminative residual images are obtained. Next, the RGB image should be transformed to more representative spaces such as HSV, YCbCr, and LAB. Dependent on the powerful texture descriptor CM, the RCTR feature is extracted from multiple color channels. Besides, an ensemble classifier is carefully designed based on a probabilistic voting rule to make the prediction. Extensive analytical experiments are conducted to verify the effectiveness of transforming color space and employing residual image. Four challenging benchmark datasets FASD, RAD, MSU, and ROSE are used to evaluate our proposed method, and our proposed RCTR-based detector shows preferable performance in the cases of both intradataset and interdataset testing.

Data Availability

All data, models, or code generated or used during the study are available from the corresponding author upon request.

Disclosure

Yuting Du and Tong Qiao are co-first authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Yuting Du and Tong Qiao contributed to the work equally.

Acknowledgments

This work was funded by the Cyberspace Security Major Program in National Key Research and Development Plan of China under grant no. 2016YFB0800201, the Natural Science Foundation of China under grant no. 61702150, the Public Research Project of Zhejiang under grant no. LGG19F020015, and the Key Research and Development Plan Project of Zhejiang Province under grant no. 2017C01065.

References

[1] L. Feng, L.-M. Po, Y. Li et al., "Integration of image quality and motion cues for face anti-spoofing: a neural network approach," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 451–460, 2016.

[2] L. Lei, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, and A. Hadid, "An original face anti-spoofing approach using partial convolutional neural network," in *Proceedings of the International Conference on Image Processing Theory Tools & Applications*, pp. 1–6, Montreal, Canada, December 2016.

[3] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," *Computer Science*, vol. 9218, pp. 373–384, 2014.

[4] N. N. Lakshminarayana, N. Narayan, N. Napp, S. Setlur, and V. Govindaraju, "A discriminative spatio-temporal mapping of face for liveness detection," in *Proceedings of the IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, Delhi, India, February 2017.

[5] Z. Xu, L. Shan, and W. Deng, "Learning temporal features using LSTM-CNN architecture for face anti-spoofing," in *Proceedings of the Asian Conference on Pattern Recognition (ACPR)*, Beijing, China, May 2015.

[6] H. Li, P. He, S. Wang, A. Rocha, X. Jiang, and A. C. Kot, "Learning generalized deep feature representation for face anti-spoofing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2639–2652, 2018.

[7] H. Li, W. Li, H. Cao, S. Wang, F. Huang, and A. C. Kot, "Unsupervised domain adaptation for face anti-spoofing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1794–1809, 2018.

[8] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, and Y. Q. Shi, "A serial image copy-move forgery localization scheme with source/target distinguishment," *IEEE Transactions on Multimedia*, vol. 25, 2015.

[9] H. Chen, G. Hu, Z. Lei, Y. Chen, N. M. Robertson, and S. Z. Li, "Attention-based two-stream convolutional networks for face spoofing detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 578–593, 2019.

[10] P. Gang, S. Lin, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, November 2017.

[11] K. Kollreider, H. Fronthaler, M. I. Faraj, and J. Bigun, "Real-time face detection and motion analysis with application in "liveness" assessment," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 548–558, 2007.

[12] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh, "Computationally efficient face spoofing detection with motion magnification," in *Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition Workshops*, Columbia, SC, USA, June 2013.

[13] B. Wei, L. Hong, L. Nan, and J. Wei, "A liveness detection method for face recognition based on optical flow field," in *Proceedings of the International Conference on Image Analysis and Signal Processing*, Marrakech, Morocco, July 2009.

[14] J. Galbally, S. Marcel, and J. Fierrez, "Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 710–724, 2014.

[15] J. Galbally and S. Marcel, "Face anti-spoofing based on general image quality assessment," in *Proceedings of the International Conference on Pattern Recognition IEEE Computer Society*, Stockholm, Sweden, August 2014.

[16] W. Di, H. Hu, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE Transactions on Information Forensics & Security*, vol. 10, no. 4, pp. 746–761, 2015.

[17] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *Proceedings of the International Joint Conference on Biometrics (IJCB)*, pp. 1–7, Washington, DC, USA, October 2011.

[18] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, "LBP-top based countermeasure against face spoofing attacks," in *Proceedigs of the Computer Vision—ACCV 2012 Workshops*, Daejeon, Korea, November 2012.

[19] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *Proceedings of the IEEE International Conference of the Biometrics Special Interest Group (BIOSIG)*, Darmstadt, Germany, September 2012.

[20] J. Komulainen, A. Hadid, and M. Pietikainen, "Context based face anti-spoofing," in *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications, and Systems (BTAS)*, Redondo Beach, CA, USA, October 2014.

[21] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face spoofing detection using colour texture analysis," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 8, pp. 1818–1830, 2016.

[22] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva, "An investigation of local descriptors for biometric spoofing

detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 849–863, 2015.

[23] S. R. Arashloo, J. Kittler, and W. Christmas, "Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 11, pp. 2396–2407, 2015.

[24] J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection with component dependent descriptor," in *Proceedings of the International Conference on Biometrics (ICB)*, pp. 1–6, Halmstad, Sweden, June 2016.

[25] B. Chen, X. Qi, Y. Zhou, G. Yang, Y. Zheng, and B. Xiao, "Image splicing localization using residual image and residual-based fully convolutional network," *Journal of Visual Communication and Image Representation*, vol. 73, Article ID 102967, 2020.

[26] P. He, X. Jiang, T. Sun, and H. Li, "Computer graphics identification combining convolutional and recurrent neural networks," *IEEE Signal Processing Letters*, vol. 25, no. 9, pp. 1369–1373, 2018.

[27] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu, "Face anti-spoofing using patch and depth-based cnns," in *Proceedings of the IEEE International Joint Conference on Biometrics*, Denver, CO, USA., April 2017.

[28] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: binary or auxiliary supervision," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CL, USA, September 2019.

[29] X. Yang, W. Luo, L. Bao et al., "Face anti-spoofing: model matters, so does data," in *Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CL, USA, September 2019.

[30] K. Ali, "Unknown presentation attack detection against rational attackers," 2010, https://arxiv.org/abs/2010.01592.

[31] H. Feng, Z. Hong, H. Yue et al., "Learning generalized spoof cues for face anti-spoofing," 2005, https://arxiv.org/abs/2005.03922.

[32] Y. Jia, J. Zhang, S. Shan, and X. Chen, "Single-side domain generalization for face anti-spoofing," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, June 2020.

[33] T. Freitas Pereira, J. Komulainen, A. Anjos et al., "Face liveness detection using dynamic texture," *Eurasip Journal on Image & Video Processing*, vol. 1, no. 2, 2014.

[34] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[35] S. G. Chang, B. Bin Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1532–1546, 2000.

[36] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[37] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[38] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Image and Signal Processing*, pp. 236–243, Springer, Berlin, Germany, 2008.

[39] R. Nosaka, Y. Ohkawa, and K. Fukui, "Feature extraction based on co-occurrence of adjacent local binary patterns," in *Advances in Image and Video Technology*Springer, Berlin, Germany, 2011.

[40] J. Kannala and E. Rahtu, "Bsif: binarized statistical image features," in *Proceedings of the Pattern Recognition (ICPR), 21st International Conference on Pattern Recognition*, Tsukuba, Japan, November 2012.

[41] A. Eleyan and H. Demirel, "Co-occurrence matrix and its statistical features as a new approach for face recognition," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 19, no. 1, pp. 97–107, 2011.

[42] X. Song, X. Zhao, L. Fang, and T. Lin, "Discriminative representation combinations for accurate face spoofing detection," *Pattern Recognition*, vol. 85, pp. 220–231, 2018.

[43] K. Patel, H. Han, and A. K. Jain, "Secure face unlock: spoof detection on smartphones," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 10, pp. 2268–2283, 2016.

[44] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face anti-spoofing using speeded-up robust features and Fisher vector encoding," *IEEE Signal Processing Letters*, vol. 24, pp. 141–145, 2017.

[45] H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data—with application to face recognition," *Pattern Recognition*, vol. 34, no. 10, pp. 2067–2070, 2001.

[46] E. Osuna, R. Freund, and F. Girosi, "Training support vector machines: an application to face detection," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 2015.

[47] T. Chen and C. Guestrin, "Xgboost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, San Francisco, CL, USA, August 2016.

[48] R. P. W. Duin, "The combining classifier: to train or not to train?" in *Proceedings of the Object Recognition Supported by User Interaction for Service Robots*, Quebec City, Canada, August 2002.

[49] Z. Zhang, J. Yan, S. Liu, L. Zhen, and S. Z. Li, "A face antispoofing database with diverse attacks," in *Proceedings of the International Conference on Biometrics (ICB)*, Seoul, Korea, August 2012.

[50] X. Tan, L. Yi, J. Liu, and J. Lin, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *Proceedings of the Computer Vision—ECCV*, Heraklion, Crete, August 2010.

[51] D. E. King, "Dlib-ml: a machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, no. 3, pp. 1755–1758, 2009.

[52] A. Swami and R. Jain, "Scikit-learn: machine learning in python," *Journal of Machine Learning Research*, vol. 12, no. 10, pp. 2825–2830, 2012.

[53] Q. T. Phan, D. T. Dang-Nguyen, G. Boato et al., "FACE spoofing detection using LDP-TOP," in *proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, Phoenix, AZ, USA, 2016.

[54] S. Tirunagari, N. Poh, D. Windridge, A. Iorliam, N. Suki, and A. T. S. Ho, "Detection of face spoofing using visual dynamics," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 762–777, 2015.

[55] A. Pinto, H. Pedrini, W. R. Schwartz, and A. Rocha, "Face spoofing detection through visual codebooks of spectral temporal cubes," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4726–4740, 2015.

[56] X. Zhao, Y. Lin, J. Heikkila et al., “Dynamic texture recognition using volume local binary count patterns with an application to 2D face spoofing detection,” *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 552–566, 2017.

[57] J. K. Z. Boulkenafet and A. Hadid, “Face anti-spoofing based on color texture analysis,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2636–2640, Quebec City, Canada, September 2015.

[58] L. Li, X. Feng, Z. Xia, X. Jiang, and A. Hadid, “Face spoofing detection with local binary pattern network,” *Journal of Visual Communication and Image Representation*, vol. 54, pp. 182–192, 2018.

[59] R. Cai, H. Li, S. Wang, C. Chen, and A. C. Kot, “DRL-fas: a novel framework based on deep reinforcement learning for face anti-spoofing,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 937–951, 2020.

[60] A. Jourabloo, Y. Liu, and X. Liu, “Face de-spoofing: anti-spoofing via noise modeling,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 290–306, Glasgow, Scotland, March 2018.

[61] D. Deb and A. K. Jain, “Look locally infer globally: a generalizable face anti-spoofing approach,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1143–1157, 2020.