Rapid #: -17460250

CROSS REF ID: **976667**

LENDER: **JBE :: Ejournals**

BORROWER: **WVU :: Downtown Campus Library**

TYPE:	Article CC:CCG
JOURNAL TITLE:	Image and vision computing
USER JOURNAL TITLE:	Image and Vision Computing
ARTICLE TITLE:	Face presentation attack detection in mobile scenarios: A comprehensive evaluation
ARTICLE AUTHOR:	Shan Jia
VOLUME:	93C
ISSUE:	
MONTH:	
YEAR:	2020
PAGES:	103826-
ISSN:	0262-8856
OCLC #:	

Processed by RapidX: 4/16/2021 7:16:15 AM

Image and Vision Computing 93 (2020) 103826

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Face presentation attack detection in mobile scenarios: A comprehensive evaluation☆

Shan Jia[a], Guodong Guo[b,*], Zhengquan Xu[a], Qiangchang Wang[b]

[a]State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China
[b]Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV 26506, USA

ARTICLE INFO

Article history:
Received 12 October 2019
Accepted 5 November 2019
Available online 12 November 2019

Keywords:
Face presentation attack
Face recognition
Performance evaluation
Biometrics

ABSTRACT

The vulnerability of face recognition systems to different presentation attacks has aroused increasing concern in the biometric community. Face presentation detection (PAD) techniques, which aim to distinguish real face samples from spoof artifacts, are the efficient countermeasure. In recent years, various methods have been proposed to address 2D type face presentation attacks, including photo print attack and video replay attack. However, it is difficult to tell which methods perform better for these attacks, especially in practical mobile authentication scenarios, since there is no systematic evaluation or benchmark of the state-of-the-art methods on a common ground (i.e., using the same databases and protocols). Therefore, this paper presents a comprehensive evaluation of several representative face PAD methods (30 in total) on three public mobile spoofing datasets to quantitatively compare the detection performance. Furthermore, the generalization ability of existing methods is tested under cross-database testing scenarios to show the possible database bias. We also summarize meaningful observations and give some insights that will help promote both academic research and practical applications.

1. Introduction

Owing to the high efficiency and accuracy in identity authentication, face recognition technology has gained rapid development and broad applications in recent years, from daily uses like smartphone unlocking and access control, to high-security applications like payment systems, e-government affairs, and counter terrorism. This popularity, however, also makes face recognition systems become a major target of spoofing attack [1] (also known as presentation attack in ISO/IEC 30107-1). An impostor may gain authorized access to an unprotected face recognition system simply by presenting a face artifact of a legitimate user, which can be easily generated based on a person's face images or videos from the 'open' social networks.

Based on the way to generate the face artifact, face presentation attacks can be classified into two categories: one including face modalities in 2D, such as printed/digital photographs and recorded videos on the mobile/tablet; another category using 3D by making a mask or presenting face models [2,3]. These types of face artifacts have been proved to be easy and effective to fool different face recognition systems. For example, the access control system without protection measures can be tricked by the photo attack (see Fig. 1 (a)). Printed photos can also fool Windows 10's Hello face authentication (see Fig. 1 (b)). Even the APPLE's iPhone X has been proved by researchers from Bkav that the Face ID can be unlocked when pointed at a 3D mask (see Fig. 1 (c)).

The vulnerability of face recognition systems (FRSs) to such presentation attacks has raised increasing concerns in recent years. Developing presentation attack detection (PAD) methods to determine whether the face at sensor level is real or fake is the efficient countermeasure. Different software-based approaches have been proposed over the last decade, which mostly focus on 2D face spoofing because of its cheap and easy implementation in practice.

We classify existing methods against 2D presentation attacks into five categories: texture based, image quality based, dynamic approaches, learned features based, and hybrid methods. Texture based schemes mainly explore the microtextural pattern differences of real faces and artifacts with the help of different texture descriptors, such as the widely used Local Binary Patterns (LBPs) [4] and Local Phase Quantization (LPQ) [5]. Image quality based methods [6,7] rely on the fact that fake faces, especially in 2D, are always vulnerable to image distortions caused by the recapture effect of faces (with paper-based photos or glass-based video screens). Dynamic approaches, exploit the temporal information to detect motion patterns across the video frames [8,9]. This kind of methods performs

☆ This paper has been recommended for acceptance by Sinisa Todorovic.

* Corresponding author.
E-mail addresses: jias@whu.edu.cn (S. Jia), guodong.guo@mail.wvu.edu (G. Guo), xuzq@whu.edu.cn (Z. Xu), qw0007@mix.wvu.edu (Q. Wang).

https://doi.org/10.1016/j.imavis.2019.11.004
0262-8856/

Fig. 1. Face recognition systems are vulnerable to presentation attacks. (a) Spoofing the access control system[1] Picture is taken from https://www.youtube.com/watch?v=1ndPi-l_f3A. (b) Microsoft system fooled by printed photo[2] Picture is taken from https://www.engadget.com/2017/12/21/windows-10-face-authentication-printed-photos-spoof/. (c) iPhone X Face ID unlocked with 3D mask[3] Picture is taken from https://www.dailydot.com/debug/face-id-mask-fail/.

well in photo attacks, but lacks robustness to replayed video attacks. Inspired by the detection success of deep learning in several vision tasks [10], learned features based PAD methods are proposed to extract adaptive features which describe trainable texture to distinguish real faces from fake ones [1]. Another trend now is to develop PAD methods based on hybrid techniques, which combine different features to benefit from the strengths of each field.

With the increasing growth of different face PAD methods, there is a critical need to overview these methods for both academia researchers and industrial developers to have a deeper understanding of the existing techniques. Some recent surveys [1,11-13] tried to summarize the advances of face presentation attack detection over the past decade. They all provided systematic analyses of the recent work on both spoofing databases and detection methodologies. In addition, three face PAD competitions have been organized [14-16] to challenge researchers to create and evaluate new counter measures on the same spoofing database. Although these surveys or competitions have approached various aspects of the state-of-the-art research in face presentation attack detection, they are still faced with the following challenges:

1. **Lack of quantitative evaluation.** Most existing surveys compare different algorithms by simply listing the reported results, without carrying out a quantitative evaluation on a common ground. Therefore, based on the results on different databases with different protocols, it is still difficult to understand how differently the existing methods can perform and which methods perform better for common 2D presentation attacks.
2. **Limited algorithms and results.** The competitions and the survey in [13] presented a common evaluation framework for comparing different detection methods on the same face spoofing database. However, the gathered methods and results on only one database are limited and not thoroughly analyzed [1].
3. **Lack of generalization ability evaluation.** These surveys or competitions, and some existing detection schemes as well, pay more attention to PAD performance based on intra-database testing or controlled environment. The robustness and generalization ability[1] have not been carefully evaluated to show how well the state-of-the-art methods can perform in more challenging conditions, such as mobile scenarios or cross-database testing scenarios, which can reflect the real-world applications by providing high-resolution images and diverse attacks.

These challenges imply that a comprehensive evaluation of the state-of-the-art PAD methods is in a high demand to establish a better understanding of different detection techniques. Therefore, the aim of this paper is to take the above three problems into consideration, and present a quantitative evaluation of several representative PAD algorithms on a common ground. Our main contributions are as follows.

- Based on the overview of the recent advances in face PAD methods, 30 representative methods from different categories are collected and re-implemented using the original codes or codes from the third party. They are quantitatively compared on a common ground, i.e., using the same databases, pre-processing operations, classifiers and evaluation metrics. This helps show what are the real differences between these methods.
- For the same PAD method, we evaluate its performance using different pre-processing operations, databases, classifiers, and evaluation metrics, to find what are the influencing factors to its performance.
- We focus on presentation attack detection in more realistic conditions. Three recently published face spoofing databases, all created in mobile scenarios under some real-world variations, are used in the experiments. Both the intra-database and cross-database testing are considered for each PAD method. This helps show what performance each face PAD method can achieve in more challenging conditions with more variations or unknown attacks.
- Some meaningful observations are obtained and summarized based on extensive experiments. We also give some deep insights into the issues that will help promote both academic research and practical applications.

The rest of the paper is organized as follows. In Section 2, we overview several representative PAD methods. Section 3 introduces the details of three mobile face spoofing databases used in our experiments. The evaluation experiments and results are presented in Section 4. We discuss the detection performance of evaluated algorithms in Section 5, followed by conclusions in Section 6.

[1] We use 'robustness' to describe the performance stability of PAD methods on different databases in known conditions, while use 'generalization ability' to describe the performance of the method against different types of unknown attacks.

2. Representative methods

Unlike hardware-based face PAD analysis, which always requires additional hardware components or user interaction, software-based methods are more efficient with low cost by designing an algorithm to tell the difference between a real face and a spoofed presentation. In this section, we first review state-of-the-art software-based face PAD methods of different categories, and then give a summary of 30 representative methods that will be evaluated in our experiments.

2.1. Texture based methods

This kind of approach is very successful in detecting face presentation attacks, mainly because it can efficiently discriminate the artifact characteristics such as the presence of pigments (due to printing defects), shade deformation (due to a display attack), and specular reflection (by the spoof medium). The most famous and widely used approaches are based on the LBP descriptor and its variations, including multi-scale LBP [4], grayscale LBP [17], local binary pattern variance (LBPV) [18], modified LBP (MLBP) [19], transitional (tLBP) and direction-coded (dLBP) [19], color LBP [20], guided scale based LBP (GSLBP) and local guided binary pattern (LGBP) [21]. These LBP based methods have shown outstanding performance in 2D type and 3D mask attack detection. The LPQ is also popular, which has a structure similar to LBP but encodes some phase information extracted through a short-time Fourier transform of the local patch, rather than gradients [22]. Different types of LPQ have achieved promising detection performance against photo and video attacks, such as multi-level LPQ (ML-LPQ) [5], multi-block LPQ (MB-LPQ) [16], and pyramid multi-level LPQ (PML-LPQ) [16]. Other texture descriptors were also exploited for presentation detection. Histogram of oriented gradients (HOG) [23-25] based methods capture the edge or gradient structures of the facial image to distinguish real faces from artifacts. Zhang et al. [26] used the Difference of Gaussian(DoG) to remove lightning variations while preserving the high frequency information. In [27], Radon transform is used to extract features about contrast, luminosity and shapes for face presentation attack detection. The gray level co-occurrence matrices (GLCM) feature also achieved discriminant global representation in detecting presentation attacks in [28]. Agarwal et al. [29] applied block-wise Haralick features for both 2D attacks and 3D mask attacks detection. Boulkenafet et al. [30] proposed a PAD method based on speeded-up robust features (SURF) and Fisher vector (FV) encoding, which yielded promising generalization capability.

Texture-based methods generally have low computation cost and perform especially well on 2D type attacks. However, their performance depends heavily on image qualities, and the generalizability remains to be improved.

2.2. Image quality based methods

Image quality analysis has achieved outstanding performance in image manipulation detection [31] of the forensic field. Face presentation attack by displaying a printed photo or replayed video, can be regarded as a type of image manipulation. Therefore, Galbally et al. [6,32] first extracted several image quality features to distinguish between real access and impostor samples. It not only achieved promising performance for both multi-biometric and multi-attack protection, but also showed high efficiency and low cost. Wen et al. [7] proposed another face PAD method based on image distortion analysis (IDA). They summarized four kinds of distortions introduced in the reflecting and capturing process, namely specular reflection, image blurriness, chromaticity distortion, and color diversity distortion. The proposed method also achieved better detection robustness and lower computational complexity than texture features.

2.3. Dynamic approaches

Dynamic methods exploit the temporal information from videos to detect the relative motion across frames. One motion pattern occurs due to the intra-face variations, such as subconscious eye blinking, head rotation, and facial muscles movements. Wei et al. [33] introduced optical flow (OF) vectors to detect subconscious head movements for face presentation attack detection. The histogram of oriented optical flow (HOOF) and histogram of magnitudes of optical flows (HMOF) were extracted to represent the facial motion directions and magnitudes in [15]. In addition, some texture-motion descriptors extracted texture features from three orthogonal planes combining spatial and temporal information, such as LBP-TOP [9], Weber Local Descriptor (WLD-TOP) [34], and Local Derivative Pattern (LDP-TOP) [35]. These methods have showed effectiveness in describing both the appearance and horizontal and vertical motion patterns in face presentation detection.

Another way is to analyze the motion consistency of the user interaction within the environment. The motion intensity between face and background regions were computed in [8,36,37] to detect photo and video attacks, which tend to have a high motion correlation.

Dynamic methods are usually highly effective in detecting photo-attacks, but they will require more computational effort in processing video sequences compared with a static approach. Besides, videos having low motion patterns or with replayed video attacks do not give good detection results.

2.4. Learned features based methods

Following a recent trend in computer vision, deep learning models are applied and trained to provide adaptive features for face presentation attack detection. Yang et al. [38] first exploited deep convolutional neural network (CNN) for face presentation attack detection using the architecture of AlexNet [39]. This method achieved remarkable improvement in 2D attacks compared with methods based on hand-crafted features. Menotti et al. [40] investigated two deep representation approaches (for architecture optimization (AO) and filter optimization (FO)) to detect presentation attacks in multi-bimetric modalities. Both 2D attacks and 3D mask attacks were considered for faces, and the results indicated the detection robustness of convolutional networks. Lucena et al. [41] detected photo, video and mask attacks based on transfer learning using a pre-trained CNN model (VGG-16). Tu and Fang [42] also proposed a fully data-driven ultra-deep model based on transfer learning. They used the pre-trained network, ResNet-50, to discover highly discriminative features, and combined it with the Long Short-Term Memory (LSTM) units to learn temporal features for classification.

These schemes are more capable of learning discriminative features in a data-driven manner to classify real faces and impostor samples. They also tend to achieve a better generalization ability for both 2D and 3D type attacks detection.

2.5. Hybrid methods

This kind of methods fuses different features at the feature level or score level to further improve the detection performance. Combining different texture features is one direct way for feature fusion. Maatta et al. [43] combined texture (LBP) and local shape features (Gabor wavelets and histogram of oriented gradients (HOG)), for printed photo attack detection. Kose et al. [44] also proposed a simple approach against photo attacks using DoG filters and LBPV features. Binarized Statistical Image Features (BSIF) and Cepstral features were combined in [46] to extract the statistical features that can capture the micro-texture variations, while densely sampled SIFT (DSIFT)

features were fused with multi-scale LBP in [47] to detect Moire Patterns caused by photo and video attacks.

Hybrid methods using texture and image quality measures also illustrate promising performance with low computational complexity. Patel et al. [48] used a fusion of LBP (effective for face texture analysis) and color moments (effective for image quality analysis), and Kim et al. [49] combined MLBP, GLCM, and image distortions analysis for robust face presentation detection. Fusion methods of motion and texture features were exploited in [37,45] using motion correlation/magnification and LBP features. These methods improved the limitations of motion-based methods but have a higher computation cost. In [16], two learned features based methods were fused with hand-crafted LBP features, showing outstanding performance for face presentation attack detection in mobile scenarios. One fine-tuned the pre-trained network SqueezeNet [50] and another was based on the Inception-v3 model [51].

Fusion of different methods is important in designing complementary and extensible countermeasures, and studying how different features can be combined to construct more effective and robust PAD frameworks, which becomes increasingly popular in recent years.

To sum up, there are both strengths and weaknesses in each category of the PAD methods. To quantitatively show the performance differences, we collect several methods to carry out evaluation on a common framework. Totally 30 methods are selected when taking the following three factors into consideration.

1) Different from the benchmark in [52], which focuses on PAD methods based solely on color texture analysis, we select diverse algorithms from all the above-mentioned categories to provide a comprehensive comparison and evaluation.

2) In each category of the methods, we try our best to include methods with outstanding detection performance or using representative feature descriptors, such as the LBP and LPQ features in texture analysis, the image distortion features in image quality based methods, the motion intensity analysis in dynamic approaches, and various combination of different features in the category of hybrid methods.

3) Considering the challenge in re-implementing various face PAD methods, especially some depth or liveness based methods (most relying on special hardwares or without original code available to the public), we limit the performance evaluation to software based approaches with the original code or the code provided by the third party. At the end, 30 methods were collected (among them, 20 with the original code). The details of these methods are summarized in Table 1.

3. Face spoofing databases

Several databases with different face presentation attacks have been proposed to promote the development of new detection schemes. Some surveys [12,13] have provided detailed information of existing face spoofing databases. However, with the increasing popularity of face recognition on mobile phones, new databases focus more on generating face presentation attacks in mobile scenarios, where the faces are captured by high-resolution cameras on modern smartphones. Therefore, we aim to carry out the evaluation on mobile databases to compare and show how well existing PAD methods can work in a more realistic condition. Taking both the database size and spoofing diversity into consideration, we select three recently released mobile spoofing databases, namely, Oulu-NPU DB, Replay-Mobile DB, and MSU-USSA DB. Table 2 provides a brief overview of these databases.

3.1. Oulu-NPU DB

This database consists of 4950 real access and presentation attack videos of 55 subjects. The videos were recorded using the front

Table 1
Brief overview of the evaluated face PAD methods.

Method	Reference	Year	Features	Attacks	Performance (classifier)	Type
A01	Anjos and Marcel [8]	2011	Motion intensity	Photo	HTER = 8.98% (MLP)	Dynamic
A02	Maatta et al. [4]	2011	Multi-scale LBP	Photo	EER = 2.90% (SVM)	Texture
A03	Chingovska et al. [17]	2012	Per-image LBP	Photo, video	HTER = 15.16% (SVM)	Texture
A04	Maatta et al. [43]	2012	LBP+Gabor+HOG	Photo	ACER = 1.10% (SVM)	Hybrid
A05	Zhang et al. [26]	2012	DoG	Photo, video	EER = 17.00% (SVM)	Texture
A06	Pereira et al. [9]	2012	LBP-TOP	Photo, video	HTER = 7.60% (LDA)	Dynamic
A07	Kose et al. [44]	2012	DoG+LBPV	Photo	EER=11.97% (chi-square dissimilarity metric)	Hybrid
A08	Bharadwaj et al. [45]	2013	Motion+multi-scale LBP	Photo, video	HTER = 3.94%[a] (SVM)	Hybrid
A09	Komulainen et al. [37]	2013	Motion+LBP	Photo, video	HTER=5.11% (Complex)	Hybrid
A10	Galbally and Marcel [6]	2014	Image quality	Photo, video	HTER = 23.80%[a] (LDA)	Quality
A11	Raghavendra and Busch [46]	2014	BSIF+Cepstral	Photo, video	ACER = 10.21% (SVM)	Hybrid
A12	Wen et al. [7]	2015	Image distortions	Photo, video	EER = 10.15%[a] (SVM)	Quality
A13	Patel et al. [47]	2015	Multi-scale LBP+DSIFT	Photo, video	HTER = 4.87%[a] (SVM)	Hybrid
A14	Benlamoudi et al. [5]	2015	ML-LPQ	Photo, video	EER = 11.39% (SVM)	Texture
A15	Boulkenafet et al. [20]	2015	Color LBP	Photo, video	EER = 3.30%[a] (SVM)	Texture
A16	Mei et al. [34]	2015	WLD-TOP	Photo, video	Accuracy=74.12%[b] (SVM)	Dynamic
A17	Albu [27]	2015	Radon transform	Photo	Accuracy = 97.20% (/)	Texture
A18	Menotti et al. [40]	2015	cf10-11 based	Photo, video, mask	HTER = 0.38% (SVM)	Learned
A19	Patel et al. [48]	2016	LBP+color moment	Photo, video	EER = 3.84% (SVM)	Hybrid
A20	Kim et al. [49]	2016	MLBP+GLCM+distortions	Photo, video	HTER = 4.28%[a] (SVM)	Hybrid
A21	Agarwal et al. [29]	2016	Haralick features	Photo, video, mask	EER = 2.03%[a] (SVM)	Texture
A22	Phan et al. [35]	2016	LDP-TOP	Photo, video	HTER = 6.04%[a] (SVM)	Dynamic
A23	Boulkenafet et al. [16]	2017	MB-LPQ	Photo, video	ACER=36.70%[c] (Softmax)	Texture
A24	Boulkenafet et al. [16]	2017	PML-LPQ	Photo, video	ACER=37.50%[c] (SVM)	Texture
A25	Boulkenafet et al. [30]	2017	Color SURF	Photo, video	EER = 1.70%[a] (Softmax)	Texture
A26	Peng et al. [21]	2017	LBP+GSLBP	Photo, video	EER = 5.54%[a] (SVM)	Hybrid
A27	Peng et al. [21]	2017	LGBP	Photo, video	EER = 4.88%[a] (SVM)	Texture
A28	Boulkenafet et al. [16]	2017	SqueezeNet based+color LBP	Photo, video	ACER=22.50%[c] (/)	Hybrid
A29	Lucena et al. [41]	2017	VGG-16 based	Photo, video, mask	HTER = 0.60%[a] (Sigmoid)	Learned
A30	Tu et al. [42]	2017	ResNet-50 based	Photo, video	HTER = 1.20%[a] (Softmax)	Learned

[a] Using the average result of different databases.
[b] Using the result on cross-database.
[c] Using the result of the most challenging protocol.

Table 2
Brief overview of used face spoofing databases.

Database	Year	#Subjects	#Real/fake	Attacks	Captured	Sample
Oulu-NPU	2017	55	990/3960	Photo, video	6 phones	Video
Replay-Mobile	2016	40	550/640	Photo, video	iPad, LG phone	Video
MSU-USSA	2016	1040	1040/8320	Photo, video	Google phone	Image

cameras of six mobile devices, including Samsung Galaxy S6 edge, HTC Desire EYE, MEIZU X5, ASUS Zenfone Selfie, Sony XPERIA C5 Ultra Dual, and OPPO N3. Three sessions with different illumination conditions and background scenes were considered to create printed photo attack (using two printers) and replayed video attack (on two display devices). The whole database was divided into three subsets for training, development, and testing, with four protocols.

3.2. Replay-Mobile DB

It contains 1190 video sequences of photo and video attack attempts to 40 clients. These videos were recorded with an iPad Mini2 and a LG-G4 smartphone under different lighting conditions. The attacks were created using two spoof mediums, fixed matte-screen and paper which was either hand-held or fixed-support.

3.3. MSU-USSA DB

This database was specifically created to simulate presentation attacks on smartphones with diversities of environment, image quality, image acquisition device. It consists of a subset (1000 subjects) of the web faces database [53] with celebrity images, and the MSU-MFSD dataset (40 subjects). This database provides both live faces and 2D presentation attacks (printed photo and replayed video attacks) in still-images. The front and rear facing cameras with different resolutions on the Google Nexus 5 was used to capture the spoofed attacks, while four kinds of spoof mediums, including MacBook, Nexus 5, Tablet screens, and paper, were used to show the live face images. Totally, 8320 images were created.

4. Evaluation

In this section, 30 representative face PAD methods are evaluated and compared following a unified framework. We first test the attack abilities of three face spoofing databases using different face recognition systems. Then we introduce the database protocols and evaluation environment. The influence of some pre-processing factors in detecting face presentation attacks are also demonstrated. After that, we evaluate and analyze the detection robustness and generalization ability of the 30 methods through intra-database and cross-database testing.

4.1. Attack abilities of face spoofing databases

We considered three FRSs to show the vulnerability towards detecting spoofed faces using the three mobile spoofing databases, so that the attack abilities of these databases can be demonstrated. For our experiments, we used a commercial system Neurotechnology VeriLook SDK [54], and two publicly available FRSs: OpenFace [55] and Face++ [56]. The Impostor Attack Presentation Match Rate (IAPMR) metric was used to report the results, which can be considered as an indication of the attack success chances if the FRS is evaluated regarding its PAD capabilities [57]. It is defined as the proportion of impostor attack presentations using the same Presentation Attack Instrument (PAI) species in which the target reference is matched in a full-system evaluation of a verification system [58]. The IAPMR values of the three FRSs on the Oulu-NPU, Replay-Mobile, and MSU-USSA databases are provided in Table 3.

Table 3 shows that over 94% of the images in the three mobile face presentation attack databases were successfully compared using the three FRSs. Lower values of IAPMR can be seen for images in the Replay-Mobile database, which is attributed to the lower image quality resulting from the recording and printing process in this database.

Table 3
IAPMR of three face recognition systems.

FRS	Threshold	Oulu-NPU	Replay-Mobile	MSU-USSA
VeriLook	36[a]	99.39%	97.61%	99.04%
Openface	0.99[b]	98.23%	94.94%	99.41%
Face++	1e-5[c]	100%	99.45%	99.57%

[a] Using the matching score when FAR=0.1%.
[b] Using a squared L2 distance threshold.
[c] Using the confidence threshold at the 0.001% error rate.

4.2. Evaluation protocols and environment

We followed the original evaluation protocols of each database to evaluate the performance of the different PAD methods (as summarized in Table 4). Three classifiers, namely, the Softmax classifier, Support Vector Machine (SVM) with linear and RBF kernels, were used to show the influence of classifiers on detection performance. Based on the ISO/IEC metrics, we reported the results on all databases using three evaluation metrics, the Attack Presentation Classification Error Rate (APCER), Bona Fide Presentation Classification Error Rate (BPCER), and the Average Classification Error Rate (ACER). They are calculated as follows:

$$APCER = \frac{1}{N_a}\sum_{i=1}^{N_a}(1 - Res_i) \tag{1}$$

$$BPCER = \frac{\sum_{i=1}^{N_r} Res_i}{N_r} \tag{2}$$

$$ACER = \frac{APCER + BPCER}{2} \tag{3}$$

where N_a is the total number of attack presentations, and N_r is the number of real samples. Res_i equals to 1 if the ith presentation is classified as an attack and 0 if classified as real. Lower values of these metrics indicate better performance of the PAD algorithms.

In addition, two recent metrics for PAD methods defined within the ISO/IEC FDIS 30107-3 [58]: the BPCER20 and BPCER10 (which represent the BPCER for a fixed APCER of 5% and 10%, respectively) were reported.

All the PAD methods were re-implemented based on the original codes or codes realized by the third party according to the description in the original papers. Most methods were evaluated under

Table 4
Evaluation protocols of used face spoofing databases.

Database	#Train	#Dev	#Test	#Protocols	Face size
Oulu-NPU	1800	1350	1800	4	64*64
Replay-Mobile	312	416	302	3	64*64
MSU-USSA	7488	/	1872	1*	120*120

* Using fivefold subject-exclusive cross validation.

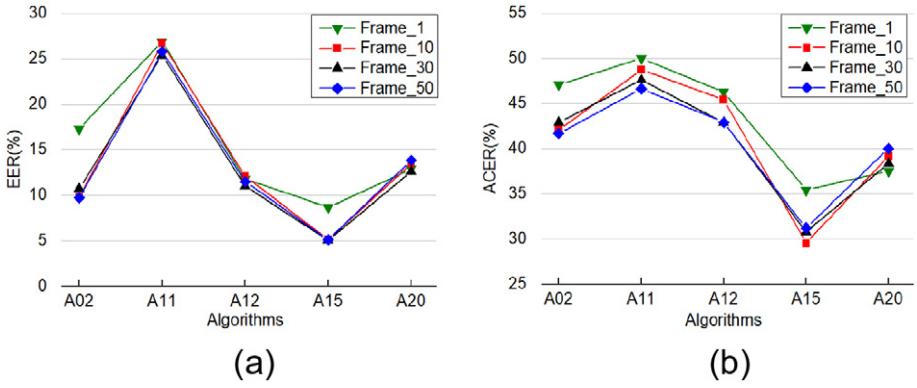

Fig. 2. Detection results under different frame numbers. (a) EER; (b) Overall ACER.

Matlab R2016b on a Windows 10 system with an Intel(R) Core(TM) i7-7500U CPU, 2.70 GHz with a 16 GB RAM. Two motion based methods (A01 and A09), image quality analysis based (A10 and A12), and four learning based methods (A18, A28-30) were run in Python 2.7 under Ubuntu Linux 16.04 LTS with an Intel(R) Core(TM) i7-6850 K CPU, 3.60GHz×12.

4.3. Influence of pre-processing

We first studied the influence of some pre-processing factors on the detection performance, including the number of frames for feature extraction, the way to select frames from videos, and the interpupillary distance (IPD) to crop face regions.

The evaluation was carried out using the Softmax classifier on the Oulu-NPU database, which provides relatively large-size video data. We randomly selected five algorithms from different categories (except the motion and learning features based methods, whose performance is not affected by the frame number or face size) as examples to show the influence. These methods are A02 (using multi-scale LBP), A11 (a hybrid method combining BSIF with Cepstral features), A12 (an image quality based method), A15 (color LBP features based), and A20 (a hybrid method combining MLBP, GLCM, and image distortions features). We illustrate the detection performance in terms of the EER in the development set and the overall ACER (corresponding to the attack with the highest APCER) in the testing set. For simplicity, only the detection results of Protocol 4 are shown, which combines the previous three protocols and is the most challenging scenario.

4.3.1. Influence of the selected frame number

We studied the effect of feature extraction from different frame numbers on face PAD performance, including randomly selected 1, 10, 30, and 50 frames. The final score for each video was computed by averaging the output scores of all frames. We cropped the faces based on the original codes[2] using the provided eye location information and IPD value (32 pixels). The results including the detection accuracy and calculation efficiency are shown in Fig. 2 and Table 5, respectively.

As shown in Fig. 2, the EER and ACER values decrease when frame number increases from 1 to 10 in most cases because more frames make the extracted features more stable. When the number of frames is 10, 30, and 50 respectively, there is little difference of the performance. However, larger frame number leads to higher computational cost. Taking both the detection accuracy and efficiency into consideration, we extract features from 10 frames in the following experiments.

4.3.2. Influence of the way to select frames

Existing PAD methods select frames from video sequences in different ways, including successive frames selection, random selection, and equal interval sampling. Therefore, we studied how these frame selection schemes can affect the detection performance. The results in Fig. 3 show that the overall performance of successive frames selection (by extracting the first 10 frames) is slightly worse than random selection and equal interval sampling (which tend to contain frames with more diversity). Considering the different length of video sequences in the database, we choose to randomly select frames for simplicity in the following experiments.

4.3.3. Influence of face size

Most published methods extract features from cropped face regions, which are always based on the eye or face location provided by databases. We varied the cropping of the facial region by altering the IPD (i.e. 24, 28, 32 and 36 pixels). Fig. 4 gives an example of the normalized face images of one subject with different IPD values.

As shown in Fig. 5, using larger IPDs to crop the face leads to smaller EER and ACER values, therefore better performance than smaller IPD values. This is because more background area can be removed while larger face region is retained when increasing the IPD values; therefore, more discriminative features can be extracted to distinguish live and spoofing images. As the faces are cropped into a square shape, to guarantee the structural integrity of faces, we use an IPD of 32 pixels to report results in all other experiments.

4.4. Robustness evaluation in mobile scenarios

With the same pre-processing operations, 30 face PAD methods were then re-implemented and evaluated on the same spoofing databases to show how well they can work in practical mobile authentication scenarios. Besides the Oulu-NPU database, two other recently published mobile spoofing databases, Replay-Mobile DB and MSU-USSA DB, were used to assess the robustness of existing algorithms.

Table 5
Calculation time* of five algorithms under different frame numbers (/s).

#Frame	A02	A11	A12	A15	A20
1	40.88	38.50	49.18	36.03	59.50
10	162.76	60.03	56.75	109.35	222.77
30	432.414	114.65	895.27	257.99	531.30
50	676.764	159.06	1868.89	408.75	845.99

* Calculation time only includes the detection process after feature extraction.

[2] https://sites.google.com/site/oulunpudatabase/.

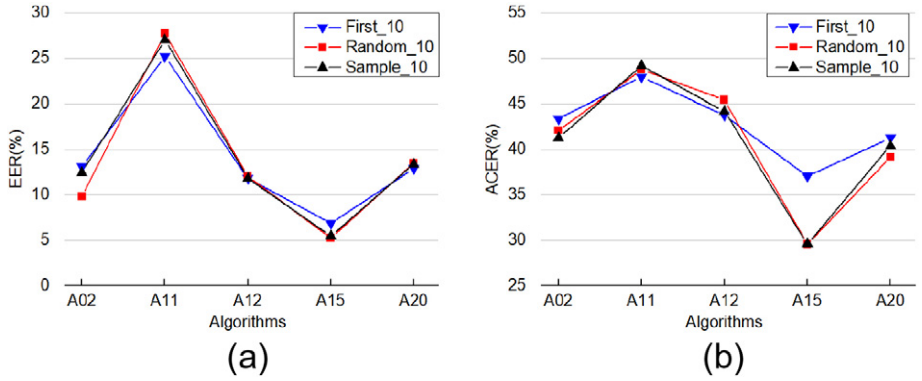

Fig. 3. Detection results with different frame selection schemes. (a) EER; (b) Overall ACER.

Fig. 4. Example of face images with different face IPDs. (a) 24 pixels; (b) 28 pixels; (c) 32 pixels; (d) 36 pixels.

4.4.1. Results on the Oulu-NPU DB

The Oulu-NPU database provides four protocols to evaluate the performances of the face PAD methods. They are designed to evaluate the effect of different environmental conditions (Protocol 1), different presentation attack instruments (PAI) (Protocol 2), different acquisition devices (Protocol 3), and combining all these variations (Protocol 4). We first list the quantitative results of Protocol 4 under Softmax classifier in Table 6 to show the detailed detection performance of 30 algorithms under the most challenging protocol.

It can be seen that the detection results vary wildly among different methods on this database. A30 based on ResNet-50 model [42] performed significantly better than other methods for both photo print and video replay attack. Besides, the learning based methods (A28, A29 and A18) and some texture based methods, including the LPQ based (A23 and A24), LBP based (A15 and A26) and Haralick features method (A21), also achieved better detection results, with the BPCER20 between 5% and 35.33%, the BPCER10 between 1.67% and 29.17%, and the overall ACER between 26.25% and 36.25%. By contrast, the overall performance was worse in some dynamic methods (A22 and A16), DoG based methods (A05 and A07), and Radon transform based method (A17), whose BPCER20 and BPCER10 were over 70% and ACER higher than 50%. For the dynamic methods, the reason for performance degradation is the low speed motion of real access videos in the Oulu-NPU database, leading to small differences from the spoofing ones, while the DoG filters used to exclude the low frequency information and noise of frames or the Radon transform used to enhance the low frequency components will perform poorly in high quality images taken on modern smartphones, which reached a similar conclusion with [26]. By comparing the ACER values in the two middle columns, we can also observe that the detection performance against replay attack is better than print attack for most algorithms. This suggests that the nature of print attacks may vary more and therefore makes it difficult to detect. In addition, the performance of the 30 algorithms reported using the BPCER20 and

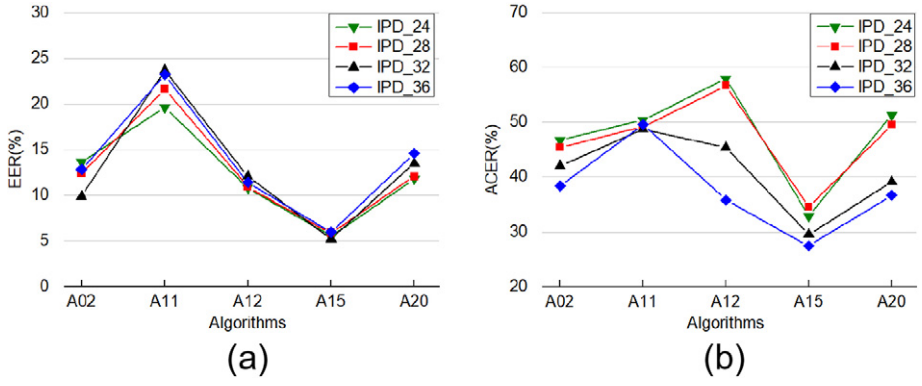

Fig. 5. Detection results with different face IPDs. (a) EER; (b) Overall ACER.

Table 6
Evaluation results (%) of Protocol 4 on Oulu-NPU DB under Softmax classifier.

Method	Features	Feature size	Dev_EER	Print	Replay	Overall			Rank
				ACER	ACER	BPCER20	BPCER10	ACER	
A01	Motion intensity [8]	1*70	31.89	36.67	31.25	67.50	56.67	**42.92**	16
A02	Multi-scale LBP [4]	10*833	9.83	40.42	33.33	63.33	50.83	**42.92**	16
A03	Per-image LBP [17]	10*59	16.72	35.41	30.00	60.00	55.00	**37.91**	12
A04	LBP+Gabor+HOG [43]	1*12225	18.50	41.25	36.67	65.83	58.33	**42.08**	15
A05	DoG [26]	10*4096	29.83	44.17	51.25	92.50	85.00	**51.25**	29
A06	LBP-TOP [9]	1*177	10.17	44.17	34.58	61.67	51.67	**47.08**	21
A07	DoG+LBPV [44]	10*59	36.28	41.67	49.17	95.83	91.67	**50.00**	24
A08	Motion+multi-scale LBP [45]	10*361	17.17	43.75	33.75	66.67	56.67	**44.58**	18
A09	Motion+LBP [37]	1*129	26.06	44.58	31.67	69.17	54.17	**47.08**	21
A10	Image quality [6]	10*18	36.39	46.25	41.67	92.50	91.67	**50.00**	24
A11	BSIF+Cepstral [46]	10*2657	29.61	45.42	46.25	89.17	89.17	**48.75**	23
A12	Image distortions [7]	10*121	12.06	43.33	32.08	60.83	55.00	**45.42**	20
A13	Multi-scale LBP+DSIFT [47]	10*4057	20.11	49.17	46.25	64.17	63.33	**50.83**	27
A14	ML-LPQ [5]	10*3587	12.33	41.67	41.25	80.83	74.17	**44.58**	18
A15	Color LBP [20]	10*354	5.06	22.92	25.83	23.33	16.67	**30.00**	5
A16	WLD-TOP [34]	1*3072	25.00	55.00	35.83	84.17	80.83	**55.00**	30
A17	Radon transform [27]	10*1800	28.67	47.91	50.41	99.17	99.17	**50.41**	26
A18	cf10-11 based [40]	10*40000	4.89	35.00	17.92	15.83	8.33	**36.25**	9
A19	LBP+color moment [48]	10*540	11.72	31.67	30.83	46.67	35.00	**36.25**	9
A20	MLBP+GLCM+distortions [49]	10*1047	13.78	35.42	30.83	60.83	53.33	**38.33**	13
A21	Haralick features [29]	10*624	10.56	30.00	25.83	5.00	1.67	**35.42**	8
A22	LDP-TOP [35]	1*21504	26.89	45.83	41.67	75.00	64.17	**50.83**	27
A23	MB-LPQ [16]	1*6912	4.56	26.25	21.67	26.67	19.17	**26.25**	2
A24	PML-LPQ [16]	1*23040	**2.44**	23.33	24.17	16.67	8.33	**30.00**	5
A25	Color SURF [30]	1*24576	6.94	35.42	30.83	13.33	8.33	**37.50**	11
A26	LBP+GSLBP [21]	10*6372	4.17	24.58	29.17	10.00	5.83	**31.67**	7
A27	LGBP [21]	10*3186	9.11	37.92	32.08	37.50	20.83	**40.00**	14
A28	SqueezeNet+color LBP [16]	10*1354	6.17	16.67	25.41	22.50	15.83	**27.92**	3
A29	VGG-16 based [41]	10*4096	15.05	25.41	25.00	35.33	29.17	**29.16**	4
A30	ResNet-50 based [42]	10*2048	3.71	**2.50**	**7.50**	**2.50**	**0.83**	**8.33**	**1**

BPECR10 show good agreement with the results reported using the ACER. However, most algorithms show higher error rates under this most challenging protocol on the Oulu-NPU database.

We further give the evaluation details on the Oulu-NPU database to show the influence of protocols and classifiers. Fig. 6 (a–c) presents the overall ACER of 30 PAD algorithms of four protocols under different classifiers. It indicates that although the ACER of different methods using the Protocol 1, 2 and 3 with only one kind of variation is generally smaller than that of Protocol 4 (the purple columns), the detection performance differences are almost consistent in four protocols. By contrast, the classifiers have a relatively large influence on the detection performance. From the average ACER curves of four protocols in Fig. 6 (d), it can be seen that for most methods, the SVM classifiers achieve lower ACER than the Softmax classifier, and the linear SVM performs slightly better than RBF-SVM classifier. Overall, the average ACER values are higher than 20% for most algorithms, except some recently published methods based on texture or deep models (A23-28 and A30), which show higher robustness in detecting face presentation attacks on this mobile database.

4.4.2. Results on the Replay-Mobile DB

The Replay-Mobile database designs three protocols for performance evaluation, namely mattescreen attack of photo and video (Protocol 1), print fixed-support and hand-held attack (Protocol 2), and a grandtest protocol for global performance evaluation (Protocol 3, which is the sum of the above attacks). Table 7 indicates the

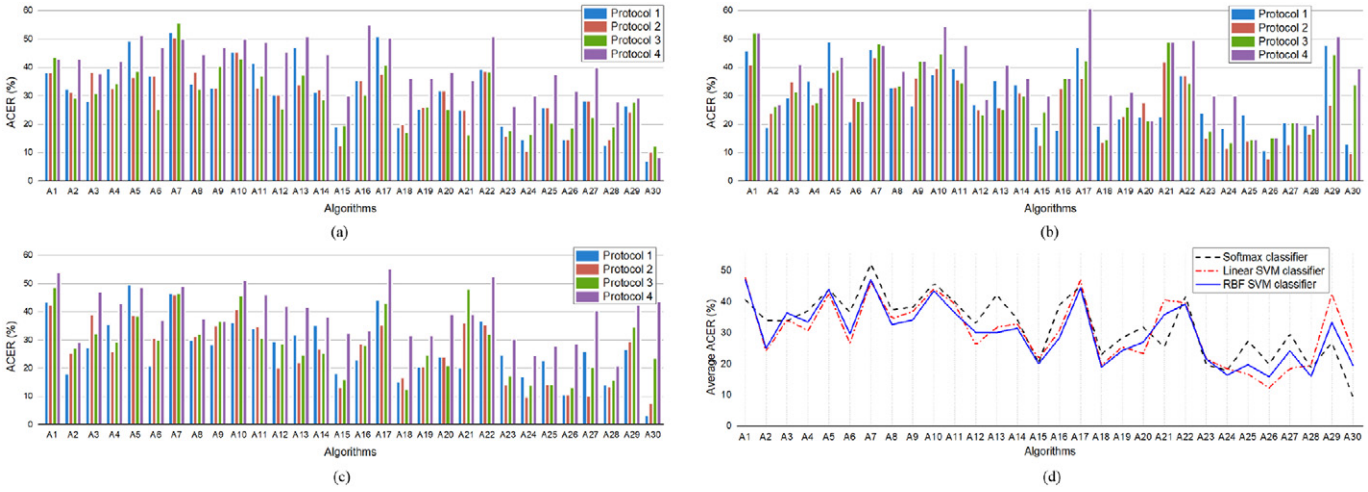

Fig. 6. Detection results of 30 algorithms on Oulu-NPU DB. (a) Overall ACER of four protocols under Softmax classifier; (b) Overall ACER of four protocols under Linear-SVM classifier; (c) Overall ACER of four protocols under RBF-SVM classifier; (d) Average ACER of all protocols under different classifiers.

Table 7
Evaluation results (%) on Replay-Mobile DB under RBF-SVM classifier.

Method	Features	Feature size	Mattescreen (ACER)		Print (ACER)		Grandtest				Rank
			Photo	Video	Fixed	Hand	Dev_EER	BPCER20	BPCER10	ACER	
A01	Motion intensity [8]	1*100	1.14	1.14	1.14	17.61	9.38	14.55	11.82	**9.55**	20
A02	Multi-scale LBP [4]	10*833	5.30	1.04	3.13	2.08	4.37	**0**	**0**	**7.03**	16
A03	Per-image LBP [17]	10*59	49.03	47.90	48.56	47.80	20.31	27.27	25.45	**20.02**	27
A04	LBP+Gabor+HOG [43]	1*12225	1.14	1.14	2.08	**0**	4.69	0.91	**0**	**0.71**	4
A05	DoG [26]	10*4096	26.04	22.82	23.86	25.47	24.37	46.36	35.45	**26.92**	29
A06	LBP-TOP [9]	1*177	7.29	2.18	3.22	4.36	7.81	5.21	4.55	**4.88**	13
A07	DoG+LBPV [44]	10*59	27.65	15.06	17.05	14.77	29.37	33.64	24.55	**18.15**	25
A08	Motion+multi-scale LBP [45]	10*361	12.59	7.39	11.74	12.50	9.37	12.73	7.27	**6.69**	15
A09	Motion+LBP [37]	1*159	2.27	1.14	1.14	10.23	6.25	15.45	9.09	**7.28**	17
A10	Image quality [6]	10*18	33.05	10.89	17.05	11.65	22.50	28.18	20.00	**16.59**	23
A11	BSIF+Cepstral [46]	10*2657	17.33	9.75	9.75	9.75	18.12	0.91	**0**	**24.22**	28
A12	Image distortions [7]	10*121	2.27	4.36	3.13	2.08	2.50	**0**	**0**	**2.54**	9
A13	Multi-scale LBP+DSIFT [47]	10*4057	38.54	26.04	34.37	31.25	7.50	**0**	**0**	**4.17**	12
A14	ML-LPQ [5]	10*3587	9.37	4.17	2.08	2.08	1.95	**0**	**0**	**2.86**	10
A15	Color LBP [20]	10*354	1.14	**0**	1.04	**0**	0	**0**	**0**	**0.26**	**1**
A16	WLD-TOP [34]	1*3072	14.96	12.78	14.20	8.43	8.20	7.27	0.91	**8.79**	19
A17	Radon transform [27]	10*1800	13.26	18.47	10.04	10.89	16.41	33.64	21.82	**10.72**	21
A18	cf10-11 based [40]	10*40000	5.68	2.08	1.04	2.27	3.52	1.82	**0**	**1.62**	5
A19	LBP+color moment [48]	10*540	4.17	3.12	4.17	2.08	2.50	**0**	**0**	**5.99**	14
A20	MLBP+GLCM+distortions [49]	10*1047	**0**	**0**	**0**	**0**	1.56	**0**	**0**	**2.34**	8
A21	Haralick features [29]	10*624	50.00	50.00	3.41	50.00	50.00	**0**	**0**	**50**	30
A22	LDP-TOP [35]	1*21504	25.76	23.48	22.06	11.84	21.09	45.45	35.45	**18.65**	26
A23	MB-LPQ [16]	1*6912	2.08	1.04	1.04	1.04	0.39	**0**	**0**	**2.02**	6
A24	PML-LPQ [16]	1*23040	1.04	**0**	**0**	**0**	1.87	1.82	**0**	**2.27**	7
A25	Color SURF [30]	1*24576	**0**	**0**	**0**	1.14	2.50	**0**	**0**	**3.13**	11
A26	LBP+GSLBP [21]	10*6372	**0**	0.17	**0**	**0**	0	**0**	**0**	**0.26**	**1**
A27	LGBP [21]	10*3186	4.26	4.17	2.18	2.18	3.75	0.91	**0**	**7.75**	18
A28	SqueezeNet+color LBP [16]	10*1354	1.13	2.27	4.54	1.14	3.13	**0**	**0**	**0.26**	**1**
A29	VGG-16 based [41]	10*4096	15.15	10.04	14.30	20.64	24.37	31.82	30.00	**17.16**	24
A30	ResNet-50 based [42]	10*2048	1.14	3.12	15.24	5.30	16.41	22.73	20.00	**14.11**	22

detection performance of 30 algorithms with all protocols under the RBF-SVM classifier.

For the Mattescreen protocol, the methods A20 combining MLBP, GLCM and image distortions, and A25 based on color SURF achieve 0% ACER for both displayed photo and video attacks. The LBP based methods (A15 and A26) also demonstrate outstanding performance. It is worth noting that compared with the results in Table 6 of the Oulu-NPU database, the performance of some dynamic methods (A01, A09, A06, and A16) improves significantly under this protocol because the capturing mobile device was supported on a fixed support when recording mattescreen attacks, so that the motion pattern is distinguishable from the real access videos with relatively large movement. Besides, the detection performance against video attack is better than photo attack as a whole. We attribute this difference to the fact that displaying the recorded videos on the mattescreen makes more difference from real accesses than showing the photo on the screen, as shown in Fig. 7.

For the Print protocol, similarly, the texture based methods A20, A24, A25, A26, and A15 are quite effective in detecting both fixed and hand print attacks, achieving around 0% ACER values. We can also observe that the ACER for hand-held attack increases obviously for the motion based methods, from 1.14% for fixed-support attack to 17.61% in A01, and from 1.14% to 10.23% in A06. This suggests that the presentation attack videos with more movement pose greater challenges to motion based detection methods.

For the Grandtest protocol, three methods A15 (using color LBP), A26 (using LBP+GSLBP), and A28 (combining SqueezeNet model and color LBP features), demonstrate the best results with the BPCER20 and BPCER10 of 0%, while ACER of only 0.26%. Besides, there is no significant differences in the detection performance of most methods,

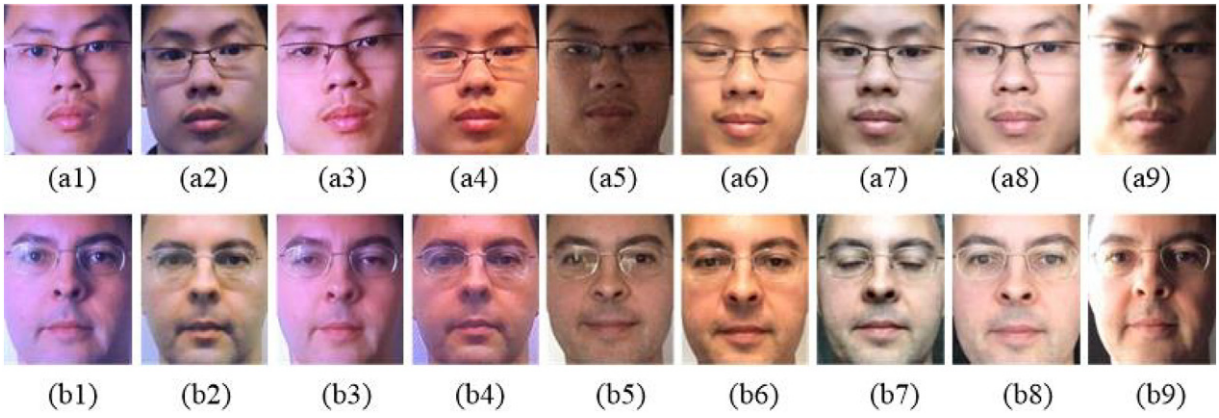

Fig. 7. Examples of cropped subject faces in the Replay-Mobile database. (a1), (b1) photo-lightoff attack; (a2), (b2) photo-lighton attack; (a3), (b3) video-lightoff attack; (a4), (b4) video-lighton attack; a(5-9), b(5-9) real accesses in different scenarios.

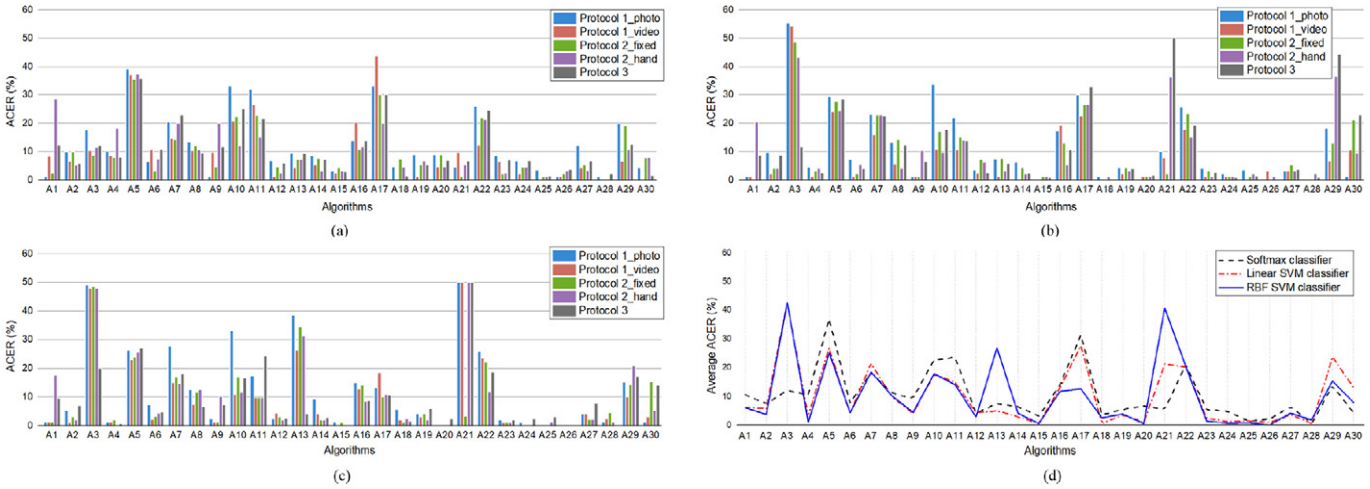

Fig. 8. Detection results of 30 algorithms on the Replay-Mobile DB. (a) Overall ACER of all protocols under the Softmax classifier; (b) Overall ACER of all protocols under the Linear-SVM classifier; (c) Overall ACER of all protocols under the RBF-SVM classifier; (d) Average ACER under different classifiers.

with 17 methods' BPCER20 and BPCER10 around 0%, and 20 methods' ACER values lower than 10.00%. We attribute this to the fact that both the dataset size and diversity of attack videos in this database are smaller than those collected in the Oulu-NPU database. Note that the method A21 seems to achieve excellent performance in terms of BPCER20 and BPCER10 (both with 0%), but using ACER metric, its performance is the worst. When we checked the APCER values corresponding to the BPCER20 and BPCER10, we found its APCERs were 100%, suggesting that the threshold for classification at APCER of 5% or 10% on the development dataset does not apply to the testing set. For this case, using the ACER metric to demonstrate the detection performance is more reasonable. Besides, the advantages of two learning based methods (A29 and A30) shown on the Oulu-NPU database are not obvious on this small-size database.

To show more details of the performance on the Replay-Mobile database, we plot column graphs of all algorithms under different protocols and classifiers in Fig. 8 (a–c). It can be observed that most ACER values for Protocol 1 with photo attack (the blue columns) are slightly larger than other attack types (because showing photo attacks on the mattescreen, especially with light on, makes less difference from real accesses, as shown in Fig. 8). However, the overall performance for different protocols is basically consistent for most algorithms, suggesting the good robustness of different methods on the same database.

Fig. 8 (d) illustrates the average ACER values of all protocols under different classifiers. Compared with results on the Oulu-NPU database in Fig. 6 (d), the Softmax classifier performs better on this database. The method A21 based on Haralick features, with the average ACER of over 40.00% under the RBF-SVM classifier, achieves about 5.00% average ACER under the Softmax classifier. The same big difference can also be observed in A03 (from over 40% with the SVM classifiers to about 10% with the Softmax classifier). These two methods also show the worst performance for both the Mattescreen and Print protocols in Table 7. The possible reason is that the smaller dataset in the Replay-Mobile database (especially for the Mattescreen and Print protocols) makes the RBF-SVM classifier sensitive

Table 8
Detection results (%) of the fivefold cross validation protocol on MSU-USSA DB under RBF-SVM classifier.

Method	Features	Feature	Overall			Rank
		size	APCER	BPCER	ACER	
A02	Multi-scale LBP [4]	1*833	6.48	6.44	**6.46**	10
A03	Per-image LBP [17]	1*59	8.81	13.17	**10.99**	16
A04	LBP+Gabor+HOG [43]	1*14441	7.61	7.50	**7.55**	11
A05	DoG [26]	1*3600	32.20	36.44	**34.32**	23
A07	DoG+LBPV [44]	1*59	27.74	28.08	**27.91**	21
A10	Image quality [6]	1*18	24.65	27.88	**26.27**	20
A11	BSIF+Cepstral [46]	1*2657	15.43	18.37	**16.90**	19
A12	Image distortions [7]	1*121	11.67	11.63	**11.65**	17
A13	Multi-scale LBP+DSIFT [47]	1*19289	7.75	7.69	**7.72**	12
A14	ML-LPQ [5]	1*3584	7.79	7.69	**7.74**	13
A15	Color LBP [20]	1*354	2.91	3.75	**3.33**	5
A17	Radon transform [27]	1*1800	32.18	37.86	**35.02**	24
A18	cf10-11 based [40]	1*40000	14.84	14.90	**14.87**	18
A19	LBP+color moment [48]	1*1602	3.32	3.27	**3.29**	4
A20	MLBP+GLCM +distortions [49]	1*1047	3.79	3.85	**3.82**	6
A21	Haralick features [29]	1*1404	7.36	8.75	**8.05**	14
A23	MB-LPQ [16]	1*6912	2.18	2.02	**2.10**	2
A24	PML-LPQ [16]	1*23040	3.05	2.98	**3.02**	3
A25	Color SURF [30]	1*24576	5.82	5.77	**5.79**	9
A26	LBP+GSLBP [21]	1*6372	1.07	0.87	**0.97**	**1**
A27	LGBP [21]	1*3186	4.92	4.81	**4.86**	8
A28	SqueezeNet+color LBP [16]	1*1354	4.87	4.81	**4.84**	7
A29	VGG-16 based [41]	1*4096	33.71	33.65	**33.68**	22
A30	ResNet-50 based [42]	1*2048	9.55	9.61	**9.58**	15

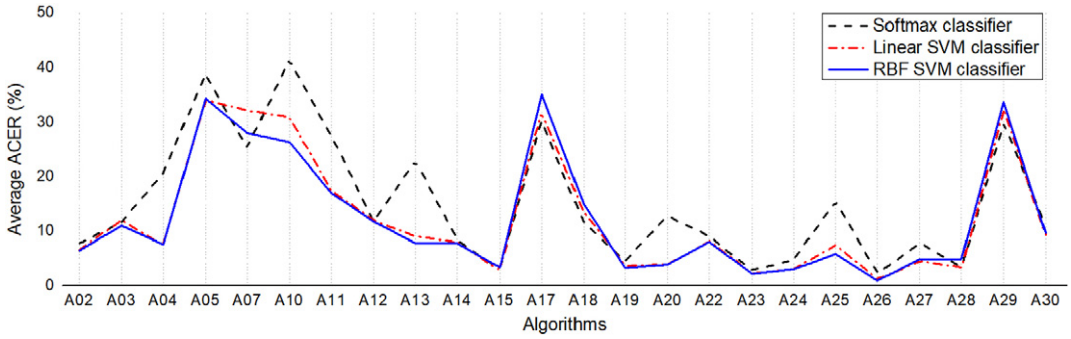

Fig. 9. Results of 24 algorithms on MSU-USSA DB under three classifiers.

to over-fitting. Overall, similarly with the results on the Oulu-NPU database, some methods based on texture or deep models (including A15, A18, A23-28 and A30) are more effective and robust against face presentation attacks on this database.

4.4.3. Results on the MSU-USSA DB

A fivefold subject-exclusive cross validation protocol is designed for MSU-USSA database. Because this is an image based database, the PAD methods based on dynamic features, including A01, A06, A08, A09, A16 and A22, are no longer applicable. Therefore, only 24 methods were evaluated on this database to show the detection differences.

Table 8 presents the results under the RBF-SVM classifier. The method combining LBP and GSLBP features (A26) indicates the best performance with ACER of 0.97%. Besides, LPQ based methods (A23 and A24) and LBP features based hybrid methods (A19, A15, A20, A28, A27) also show impressive detection performance, with ACER less than 5.00%. In Fig. 9 , the average ACER curves under different classifiers demonstrate slightly better results of two SVM classifiers than the Softmax classifier. The overall ACER values under different classifiers are lower than 20% for most methods due to the high-resolution images in this database.

4.5. Generalization ability in cross-database testing scenarios

Intra-database testing (with training and testing data captured in the same scenarios) cannot represent all real world conditions [59]. To further show the generalization ability in detecting unknown attacks, we conducted a series of cross-database experiments in this section. Each detection method was trained on one database, and tested on a different one. The results including both intra-database and cross-database testing under the Softmax classifier are shown in Table 9.

It can be seen clearly that the detection performance decreases dramatically for most algorithms when dealing with unknown

Table 9
Performance of cross-database testing on Oulu-NPU DB, Replay-Mobile DB and MSU-USSA DB under the Softmax classifier. Performance reported in terms of ACER (%).

Method	Features	Oulu-NPU			Replay-Mobile			MSU-USSA			Average	Rank
		Oulu	Replay	MSU	Replay	Oulu	MSU	MSU	Oulu	Replay		
A01	Motion intensity [8]	28.40	**2.27**	/	11.64	61.15	/	/	/	/	31.71	4
A02	Multi-scale LBP [4]	21.98	36.15	24.64	5.85	40.10	36.09	7.42	44.48	36.67	36.36	6
A03	Per-image LBP [17]	24.97	39.08	33.83	12.61	38.85	29.39	11.06	43.02	41.29	37.58	8
A04	LBP+Gabor+HOG [43]	25.24	45.38	51.44	7.28	43.16	53.85	23.05	52.50	58.26	50.76	27
A05	DoG [26]	34.97	50.72	49.19	30.10	49.69	40.05	46.81	50.24	53.38	48.88	26
A06	LBP-TOP [9]	18.92	45.64	/	12.61	41.35	/	/	/	/	43.50	18
A07	DoG+LBPV [44]	46.11	62.61	40.20	19.90	51.39	51.44	24.49	44.69	58.00	51.39	28
A08	Motion+multi-scale LBP [45]	27.74	37.13	/	9.23	43.85	/	/	/	/	40.49	15
A09	Motion+LBP [37]	27.74	50.00	/	13.59	59.97	/	/	/	/	54.98	30
A10	Image quality [6]	39.20	48.57	41.53	24.06	55.17	52.85	31.98	39.51	45.90	47.26	24
A11	BSIF+Cepstral [46]	30.80	43.69	53.34	21.85	46.08	59.68	29.09	31.22	41.74	45.96	20
A12	Image distortions [7]	15.56	16.25	33.32	2.67	42.47	42.52	13.43	35.90	14.30	30.79	2
A13	Multi-scale LBP+DSIFT [47]	26.98	41.74	46.69	7.28	38.58	21.15	17.76	47.74	47.33	40.54	16
A14	ML-LPQ [5]	23.72	41.29	35.70	7.02	39.17	37.35	7.90	43.13	39.34	39.33	12
A15	Color LBP [20]	7.50	45.38	39.03	1.69	36.35	32.90	3.34	40.28	34.72	38.11	9
A16	WLD-TOP [34]	23.85	31.08	/	14.56	46.22	/	/	/	/	38.65	10
A17	Randon transform [27]	32.74	40.77	42.85	27.70	49.41	40.50	30.41	48.06	61.18	47.13	23
A18	cf10-11 based [40]	9.41	44.93	40.02	1.69	46.63	58.32	11.51	36.67	43.43	46.13	21
A19	LBP+color moment [48]	15.83	20.87	41.32	4.88	43.85	56.22	12.95	53.02	43.95	43.21	17
A20	MLBP+GLCM+distortions [49]	15.56	26.20	32.15	4.88	46.22	49.49	11.99	41.74	36.63	38.74	11
A21	Haralick features [29]	10.80	51.95	58.26	5.33	53.96	52.49	8.74	32.53	67.95	52.86	29
A22	LDP-TOP [35]	34.41	37.13	/	22.11	42.57	/	/	/	/	39.85	13
A23	MB-LPQ [16]	18.30	39.79	58.86	2.21	45.83	36.60	**2.31**	**19.58**	15.00	35.95	5
A24	PML-LPQ [16]	12.67	36.67	41.56	1.24	40.52	38.61	4.03	46.39	57.28	43.51	19
A25	Color SURF [30]	12.60	52.21	46.91	0.98	43.09	45.61	17.01	50.24	39.79	46.31	22
A26	LBP+GSLBP [21]	9.97	43.69	55.50	3.38	**35.87**	**4.81**	2.46	37.53	42.72	36.69	7
A27	LGBP [21]	9.69	37.13	32.81	7.28	36.91	7.93	10.64	39.31	31.34	30.90	3
A28	SqueezeNet+color LBP [16]	7.74	36.67	35.19	1.24	49.24	41.80	3.34	37.19	40.77	40.14	14
A29	VGG-16 based [41]	20.80	43.43	51.89	12.16	48.02	54.48	32.21	49.41	41.74	48.16	25
A30	ResNet-50 based [42]	**7.19**	28.87	**22.09**	**0.91**	47.47	41.53	10.00	24.41	**10.66**	**29.17**	**1**

attack scenarios. Specifically, for the same method, the ACER values increase more significantly when using the Replay-Mobile or MSU-USSA database as training set. The reason is that these two databases contain less variations in the collected data than the Oulu-NPU database. Therefore, the models optimized for these databases are not able to generalize well in new acquisition conditions. This also explains why the ACER values using the Oulu-NPU database as testing are always higher. We notice one exception is that the cross-dataset testing performance of A01 (trained on the Oulu-NPU and tested on the Replay-Mobile database) improves significantly instead of degrading. This is because when training this motion intensity based method on the Oulu-NPU database, whose video frames have smaller motion amplitudes, the method can easily detect presentation attack videos with obvious movement, such as videos in the Replay-Mobile database. Otherwise (trained on the Replay-Mobile and tested on the Oulu-NPU database), its errors will increase sharply, from 11.64% ACER to 61.15% in the experiment.

For different methods, we notice that the detection performance varies widely in cross-dataset testing scenarios, ranging from 2.17% to 67.95%. To compare the generalization ability of different methods more clearly, we averaged the ACER values of the three groups of cross-database testing results in the second-to-last column. It can be seen that the average ACER values are all between 29.17% and 54.39%. Specifically, the method A30 based on ResNet-50 model, A12 using image distortions, A01 based on motion intensity, and most LBP based methods show a relatively better generalization ability. But no single algorithm can work equally well in different cross-dataset testing scenarios.

5. Discussion

Based on the evaluation results on unified evaluation frameworks in Section 4, we summarize the main observations and give some deep insights into the face presentation attack detection.

5.1. Detection performance

From the intra-database testing results on three face spoofing databases, we can observe that the performance of most methods in mobile scenarios was not as good as the reported results shown in Table 1, suggesting the poor robustness in more realistic conditions. For attacks in cross-database testing, these methods also showed unstable performance based on different training datasets.

Overall, some texture features, especially the LBP based (A26, A20, A27, A19, and A15), the LPQ (A24 and A23) and color SURF (A25) based, showed powerful abilities to distinguish real faces from artifacts. Two learning based methods, the A30 using ResNet-50 model, and A28 combining SqueezeNet model features with color LBP, also demonstrated promising potentials for face presentation attack detection. By contrast, quality based (A10 and A12) and dynamic methods (A01, A06, A16, and A22) performed worse. The reasons behind this performance difference are analyzed as follows.

- The superior performance of LBP and LPQ based PAD methods benefits from the features' highly discriminative power in local texture description. The LBP feature has the ability to code fine details by computing the gradient directions of images, and the resistance to lighting variations due to the invariance to monotonic gray-scale changes [60]. As a family of LBP-based detectors, the LPQ shares some similar advantages with LBP, which is more robust to blur variation. Therefore, in face presentation attacks, the artifact characteristics caused by printed/digital photographs or recorded videos on the mobile/tablet can be detected by using these micropattern texture desciptors. In addition, because the color gamut of printing and display devices to create the attacks is limited [30], exploiting the intrinsic disparities in the color texture also helps discriminate real from fake faces, especially in the HSV and YCbCr spaces (whose luminance and chrominance information are separated and more stable). This leads to the more robust and generalized detection performance of the color analysis based methods (A15 and A25).
- For the learning based methods, the A29 using the VGG-16 model performed worse than other data-driven based methods. Since the VGG-16 model has much more parameters (134.25 million) than models ResNet-50 (23.51 million) and SqueezeNet (1.19 million), it tends to have the overfitting problem towards small datasets, especially on the Replay-Mobile and MSU-USSA databases in our experiments.
- The three face presentation attack databases used in the experiments all consist of high-resolution and small-motion spoofing videos in mobile scenarios. Therefore, for image quality based or dynamic methods, the quality or motion differences between real accesses and spoofed images are more difficult to discern.

To sum up, the performance evaluation indicates the potential of using robust local micropattern or separated color spaces based texture descriptors to detect face presentation attacks. Also, some deep models with less parameters to be fine-tuned trend to achieve better results in existing small-size face spoofing databases. For the poor generalization ability of existing methods in detecting unknown attacks (the best average ACER in cross-database testing is about 30%, which is far away from the requirement in practical applications), one potential solution is to use a joint training strategy combining data of multiple databases to reduce the database biases [59]. It is also suggested in [38] to adapt learned models to new data based on transfer learning to improve inevitable biases among different datasets.

5.2. Databases

Experimental results also show the influence of databases on the detection performance. Both the database size and attack diversity play an important role in designing and evaluating the PAD schemes. Limited number of samples and types of attacks will not only weaken the detection performance in practical applications, but also limit the detection ability of data-driven-based methods, such as deep learning based methods, which may not have enough data for training CNNs by fine-tuning the pretrained models to their full potential [16,61]. The database diversity can be enhanced by using different input sensors, printers and display devices, and different acquisition environment (as the Oulu-NPU database did), using different lighting conditions, and motion patterns (as the Replay-Mobile database did), enhancing the subject diversity (as the MSU-USSA dababase did), and including more types of attack (such as the challenging 3D mask presentation attacks). However, there are no such comprehensive, large-scale and diverse databases yet, which are in high demand to reflect the real-world situations, and help promote more practical and generalized PAD methods.

5.3. Evaluation metrics

Based on the APCER and BPCER metrics, we reported the detection performance using the ACER and BPCER20 and BPCER10. From the results in Tables 6 and 7, we observe that these two kinds of metrics show good agreement for most algorithms, but exceptions may occur, which will result in significant performance differences for the same PAD method. There are cases that a lower BPCER20 or BPCER10 may result from higher APCER values (see A21 in Table 7), while a lower ACER may come from the unbalanced APCER and BPCER values

(see A11, A14 in Table 6). Therefore, we emphasize the need to evaluate and report the detection performance based on multiple metrics, characterizing the methods from different aspects.

5.4. Other influencing factors

We also found that the pre-processing operations, database protocols, and classifiers all have the impact on the detection results. To sum up, selecting an appropriate number of frames (10 frames are preferred in the experiments) to extract features, using larger IPD to crop faces, and applying the SVM classifiers for databases with a larger size while Softmax classifier in smaller databases, can contribute more to a better detection performance.

6. Conclusion

To have a deep understanding of the research and development in face presentation attack detection, we present a comprehensive evaluation of the state-of-the-art face PAD methods on a common ground. Totally 30 methods have been re-implemented and evaluated in three mobile spoofing databases with high-resolution images and real-world variations. Through the intra-database and cross-database testing, the detection robustness for known attacks and generalization ability for unknown attacks have been compared and analyzed. Experimental results show that most detection methods suffered from performance degradation in mobile scenarios. Although some texture features and learning based features show outstanding performance, the results in more realistic cross-database testing scenarios are far from satisfactory. Therefore, we highlight the importance of collecting more large-scale and high-diversity databases, and developing more practical and generalized PAD methods to address the database bias problems in future work.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the CMVS at the University of Oulu, the Idiap Research Institute, and MSU's PRIP Lab for providing the datasets for our evaluation. This work is partly supported by an NSF-CITeR project, and Applied Basic Research Program of Wuhan (no. 2017010201010114).

References

[1] L. Souza, L. Oliveira, M. Pamplona, J. Papa, How far did we get in face spoofing detection? Eng. Appl. Artif. Intel. 72 (2018) 368–381.
[2] S. Jia, G. Guo, Z. Xu, A survey on 3D mask presentation attack detection and countermeasures, Pattern Recogn. 98 (2020) 107032.
[3] S. Jia, C. Hu, G. Guo, Z. Xu, A database for face presentation attack using wax figure faces, International Conference on Image Analysis and Processing, Springer. 2019, pp. 39–47.
[4] J. Määttä, A. Hadid, M. Pietikäinen, Face spoofing detection from single images using micro-texture analysis, Biometrics (IJCB), 2011 international joint conference on, IEEE. 2011, pp. 1–7.
[5] A. Benlamoudi, D. Samai, A. Ouafi, S. Bekhouche, A. Taleb-Ahmed, A. Hadid, Face spoofing detection using multi-level local phase quantization (ML-LPQ), Proc. of the First Int. Conf. on Automatic Control, Telecommunication and signals ICATS15, 2015.
[6] J. Galbally, S. Marcel, Face anti-spoofing based on general image quality assessment, Pattern Recognition (ICPR), 2014 22nd International Conference on, IEEE. 2014, pp. 1173–1178.
[7] D. Wen, H. Han, A.K. Jain, Face spoof detection with image distortion analysis, IEEE Trans. Inf. Forensic Secur. 10 (4) (2015) 746–761.
[8] A. Anjos, S. Marcel, Counter-measures to photo attacks in face recognition: a public database and a baseline, Biometrics (IJCB), 2011 international joint conference on, IEEE. 2011, pp. 1–7.
[9] T. de Freitas Pereira, A. Anjos, J.M. De Martino, S. Marcel, LBP-TOP based countermeasure against face spoofing attacks, Asian Conference on Computer Vision, Springer. 2012, pp. 121–132.
[10] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.
[11] D.R. Kisku, R.D. Rakshit, Face spoofing and counter-spoofing: a survey of state-of-the-art algorithms, Trans. Mach. Learn. Artif. Intell. 5 (2) (2017) 31.
[12] J. Galbally, S. Marcel, J. Fierrez, Biometric antispoofing methods: a survey in face recognition., IEEE Access 2 (1530-1552) (2014) 1.
[13] R. Ramachandra, C. Busch, Presentation attack detection methods for face recognition systems: a comprehensive survey, ACM Comput. Surv. (CSUR) 50 (1) (2017) 8.
[14] M.M. Chakka, A. Anjos, S. Marcel, R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori, et al. Competition on counter measures to 2-d facial spoofing attacks, Biometrics (IJCB), 2011 International Joint Conference on, IEEE. 2011, pp. 1–6.
[15] I. Chingovska, J. Yang, Z. Lei, D. Yi, S.Z. Li, O. Kahm, C. Glaser, N. Damer, A. Kuijper, A. Nouak, et al. The 2nd competition on counter measures to 2D face spoofing attacks, Biometrics (ICB), 2013 International Conference on, IEEE. 2013, pp. 1–6.
[16] Z. Boulkenafet, J. Komulainen, Z. Akhtar, A. Benlamoudi, D. Samai, S. Bekhouche, A. Ouafi, F. Dornaika, A. Taleb-Ahmed, L. Qin, et al. A competition on generalized software-based face presentation attack detection in mobile scenarios, IJCB 7 (2017).
[17] I. Chingovska, A. Anjos, S. Marcel, On the effectiveness of local binary patterns in face anti-spoofing, Proceedings of the 11th International Conference of the Biometrics Special Interes Group, 2012.EPFL-CONF-192369.
[18] N. Kose, J.-L. Dugelay, Classification of captured and recaptured images to detect photograph spoofing, Informatics, Electronics & Vision (ICIEV), 2012 International Conference on, IEEE. 2012, pp. 1027–1032.
[19] N. Erdogmus, S. Marcel, Spoofing 2D face recognition systems with 3D masks, Biometrics Special Interest Group (BIOSIG), 2013 International Conference of the, IEEE. 2013, pp. 1–8.
[20] Z. Boulkenafet, J. Komulainen, A. Hadid, Face anti-spoofing based on color texture analysis, Image Processing (ICIP), 2015 IEEE International Conference on, IEEE. 2015, pp. 2636–2640.
[21] F. Peng, L. Qin, M. Long, Face presentation attack detection using guided scale texture, Multimed. Tool Appl. (2017) 1–27.
[22] D. Gragnaniello, G. Poggi, C. Sansone, L. Verdoliva, An investigation of local descriptors for biometric spoofing detection, IEEE Trans. Inf. Forensic Secur. 10 (4) (2015) 849–863.
[23] J. Komulainen, A. Hadid, M. Pietikainen, Context based face anti-spoofing, Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on, IEEE. 2013, pp. 1–8.
[24] J. Yang, Z. Lei, S. Liao, S.Z. Li, Face liveness detection with component dependent descriptor., ICB 1 (2013) 2.
[25] J. Yang, Z. Lei, D. Yi, S.Z. Li, Person-specific face antispoofing with subject domain adaptation, IEEE Trans. Inf. Forensic Secur. 10 (4) (2015) 797–809.
[26] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, S.Z. Li, A face antispoofing database with diverse attacks, Biometrics (ICB), 2012 5th IAPR International Conference on, IEEE. 2012, pp. 26–31.
[27] R.D. Albu, Face anti-spoofing based on Radon transform, Engineering of Modern Electric Systems (EMES), 2015 13th International Conference on, IEEE. 2015, pp. 1–4.
[28] A. Pinto, W.R. Schwartz, H. Pedrini, A. de Rezende Rocha, Using visual rhythms for detecting video-based facial spoof attacks, IEEE Trans. Inf. Forensic Secur. 10 (5) (2015) 1025–1038.
[29] A. Agarwal, R. Singh, M. Vatsa, Face anti-spoofing using Haralick features, Biometrics Theory, Applications and Systems (BTAS), 2016 IEEE 8th International Conference on, IEEE. 2016, pp. 1–6.
[30] Z. Boulkenafet, J. Komulainen, A. Hadid, Face antispoofing using speeded-up robust features and Fisher vector encoding, IEEE Signal Proc. Lett. 24 (2) (2017) 141–145.
[31] S. Bayram, I. Avcibas, B. Sankur, N.D. Memon, Image manipulation detection, J. Electron. Imaging 15 (5) (2006) 041102.
[32] J. Galbally, S. Marcel, J. Fierrez, Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition, IEEE Trans. Image Process. 23 (2) (2014) 710–724.
[33] W. Bao, H. Li, N. Li, W. Jiang, A liveness detection method for face recognition based on optical flow field, Image Analysis and Signal Processing, 2009. IASP 2009. International Conference on, IEEE. 2009, pp. 233–236.
[34] L. Mei, D. Yang, Z. Feng, J. Lai, WLD-TOP based algorithm against face spoofing attacks, Chinese Conference on Biometric Recognition, Springer. 2015, pp. 135–142.
[35] Q.-T. Phan, D.-T. Dang-Nguyen, G. Boato, F.G. De Natale, FACE spoofing detection using LDP-TOP, Image Processing (ICIP), 2016 IEEE International Conference on, IEEE. 2016, pp. 404–408.
[36] A. Anjos, M.M. Chakka, S. Marcel, Motion-based counter-measures to photo attacks in face recognition, IET Biom. 3 (3) (2013) 147–158.
[37] J. Komulainen, A. Hadid, M. Pietikäinen, A. Anjos, S. Marcel, Complementary countermeasures for detecting scenic face spoofing attacks, Biometrics (ICB), 2013 International Conference on, IEEE. 2013, pp. 1–7.
[38] J. Yang, Z. Lei, S.Z. Li, Learn convolutional neural network for face anti-spoofing, arXiv preprint. (2014) arXiv:1408.5601.

[39] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems, 2012. pp. 1097–1105.
[40] D. Menotti, G. Chiachia, A. Pinto, W.R. Schwartz, H. Pedrini, A.X. Falcao, A. Rocha, Deep representations for iris, face, and fingerprint spoofing detection, IEEE Trans. Inf. Forensic Secur. 10 (4) (2015) 864–879.
[41] O. Lucena, A. Junior, V. Moia, R. Souza, E. Valle, R. Lotufo, Transfer learning using convolutional neural networks for face anti-spoofing, International Conference Image Analysis and Recognition, Springer. 2017, pp. 27–34.
[42] X. Tu, Y. Fang, Ultra-deep neural network for face anti-spoofing, International Conference on Neural Information Processing, Springer. 2017, pp. 686–695.
[43] J. Määttä, A. Hadid, M. Pietikäinen, Face spoofing detection from single images using texture and local shape analysis, IET Biom. 1 (1) (2012) 3–10.
[44] N. Kose, J.-L. Dugelay, Classification of captured and recaptured images to detect photograph spoofing, Informatics, Electronics & Vision (ICIEV), 2012 International Conference on, IEEE. 2012, pp. 1027–1032.
[45] S. Bharadwaj, T.I. Dhamecha, M. Vatsa, R. Singh, Computationally efficient face spoofing detection with motion magnification, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2013. pp. 105–110.
[46] R. Raghavendra, C. Busch, Presentation attack detection algorithm for face and iris biometrics, Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European, IEEE. 2014, pp. 1387–1391.
[47] K. Patel, H. Han, A.K. Jain, G. Ott, Live face video vs. spoof face video: use of moiré patterns to detect replay video attacks, Biometrics (ICB), 2015 International Conference on, IEEE. 2015, pp. 98–105.
[48] K. Patel, H. Han, A.K. Jain, Secure face unlock: spoof detection on smartphones, IEEE Trans. Inf. Forensic Secur. 11 (10) (2016) 2268–2283.
[49] I. Kim, J. Ahn, D. Kim, Face spoofing detection with highlight removal effect and distortions, Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on, IEEE. 2016, pp. 004299–004304.
[50] F.N. Iandola, S. Han, M.W. Moskewicz, K. Ashraf, W.J. Dally, K. Keutzer, Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size, arXiv preprint. (2016) arXiv:1602.07360.
[51] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016. pp. 2818–2826.
[52] Z. Boulkenafet, J. Komulainen, A. Hadid, On the generalization of color texture-based face anti-spoofing, Image Vis. Comput. 77 (2018) 1–9.
[53] D. Wang, S.C. Hoi, Y. He, J. Zhu, T. Mei, J. Luo, Retrieval-based face annotation by weak label regularized local coordinate coding, IEEE Trans. Pattern. Anal. Mach. Intell. 36 (3) (2014) 550–563.
[54] Neurotechnology, VeriLook SDK, http://www.neurotechnology.com/verilook.html, Accessed date: 12 November 2019.
[55] B. Amos, B. Ludwiczuk, M. Satyanarayanan, et al. Openface: a general-purpose face recognition library with mobile applications, CMU Sch. Comput. Sci. (2016).
[56] Face++, Face Compare SDK, https://www.faceplusplus.com/face-compare-sdk/, Accessed date: 12 November 2019.
[57] U. Scherhag, R. Raghavendra, K.B. Raja, M. Gomez-Barrero, C. Rathgeb, C. Busch, On the vulnerability of face recognition systems towards morphed face attacks, 2017 5th International Workshop on Biometrics and Forensics (IWBF), IEEE. 2017, pp. 1–6.
[58] I.O. for Standardization, Information Technology-Biometric presentation attack detection-Part 3: Testing and Reporting, JTC 1/SC 37, Geneva, Switzerland, ISO/IEC FDIS 30107-3:2017, 2017.
[59] T. de Freitas Pereira, A. Anjos, J.M. De Martino, S. Marcel, Can face anti-spoofing countermeasures work in a real world scenario?, Biometrics (ICB), 2013 International Conference on, IEEE. 2013, pp. 1–8.
[60] L. Nanni, A. Lumini, S. Brahnam, Survey on LBP based texture descriptors for image classification, Expert Syst. Appl. 39 (3) (2012) 3634–3641.
[61] Q. Wang, Y. Zheng, G. Yang, W. Jin, X. Chen, Y. Yin, Multiscale rotation-invariant convolutional neural networks for lung texture classification, IEEE J. Biomed. Health Inform. 22 (1) (2018) 184–195.