Live Face Video vs. Spoof Face Video: Use of Moiré Patterns to Detect Replay Video Attacks

Keyurkumar Patel[†], *Hu Han*[†], *Anil. K. Jain*[†] *and Greg Ott*[‡]
Michigan State University[†]
{patelke6, hhan, jain}@msu.edu
KEYW Corporation[‡]
gott@keywcorp.com

Abstract

With the wide deployment of face recognition systems in applications from border control to mobile device unlocking, the combat of face spoofing attacks requires increased attention; such attacks can be easily launched via printed photos, video replays and 3D masks. We address the problem of facial spoofing detection against replay attacks based on the analysis of aliasing in spoof face videos. The application domain of interest is mobile phone unlock. We analyze the moiré pattern aliasing that commonly appears during the recapture of video or photo replays on a screen in different channels (R, G, B and grayscale) and regions (the whole frame, detected face, and facial component between the nose and chin). Multi-scale LBP and DSIFT features are used to represent the characteristics of moiré patterns that differentiate a replayed spoof face from a live face (face present). Experimental results on Idiap replay-attack and CASIA databases as well as a database collected in our laboratory (RAFS), which is based on the MSU-FSD database, shows that the proposed approach is very effective in face spoof detection for both cross-database, and intra-database testing scenarios.

1. Introduction

With the widespread use of smartphones, biometric systems, such as face and fingerprint recognition, are becoming increasingly popular to use as an authentication method. Two of the most popular mobile operating systems, Android and iOS, use face and fingerprint, respectively to authenticate users. With the release of Android 4.0 (Ice Cream Sandwich), Android allows users to unlock their smartphone via facial recognition (FR) technology; on all iPhones released after the iPhone 5c, iOS allows users to unlock their smartphone with their fingerprint (Touch ID). As the use of biometrics for smartphone unlocking and user authentication continues to increase, capabilities to detect spoof biometric attacks are needed to alleviate user concerns. Spoof biometric attacks launched against a smartphone's authentication system may allow malicious users to gain access to the smartphone. These attacks may lead to dire consequences, including the leakage of sensitive private data such as bank information via apps like Google Wallet and Apple Pay.[1]

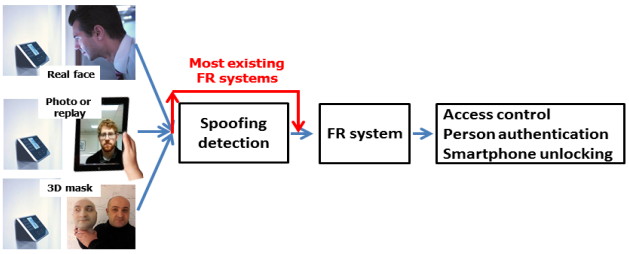


Figure 1. A face recognition (FR) with spoofing detection module. Most FR systems either do not currently have this module or this module does not perform effectively.

Given the prevalence of high resolution face images shared, (often publicly) through social media, it is relatively easy to obtain a spoof face of a user and launch a spoof attack against FR systems (see Fig. 1[2]). Compared to attacks against fingerprint recognition systems, the ubiquitous nature of image acquisition devices, such as cameras, and smartphones, allows attackers to acquire facial images of a user easily and discretely. Spoof attacks against FR systems mainly consist of (i) printed photo attack, (ii) replay attack, and (iii) 3D facial mask attack.

In this paper, we focus on video replay attacks (display a video or photo on a screen) because these attacks are easier to launch than either printed photo attack or 3D facial mask attack. Printed photo attacks require the use of high quality 2D printers and 3D facial mask attacks require high

[1] 50% of McDonald's tap-to-pay transactions are done with Apple Pay: www.ubergizmo.com/2014/11/apple-pay-accounts-for-50-of-mcdonalds-tap-to-pay-transactions/

[2] Images from: www.oulu.fi/infotech/annual_report/2013/cmv

Table 1. A summary of published methods on face spoof detection.

Method	Strength	Limitation	Replay state of the art performance (HTER)†
Face image analysis [4, 13, 11, 2, 6, 9]	Some of the methods have low computational cost and fast response	Poor generalizability, Requires face and/or landmark detection	**Idiap** (Intra-DB, Cross-DB) (1.3% [4], 47.1% [14]) **CASIA** (Intra-DB, Cross-DB) (11.8% [18], 48.3% [14])
Image quality analysis [7], [Proposed]	Good generalizability, Low computational cost, Fast response time, Face and/or landmark detection not required	Some of the image quality measures can be device dependent	**Idiap** (Intra-DB, Cross-DB) [7]: (15.2%, n/a) Proposed: (3.3%, 18.0%) **CASIA** (Intra-DB, Cross-DB) Proposed: (0.0% , 49.0%) **RAFS** (Intra-DB, Cross-DB) Proposed: (10.9% , 11.4%)

†Half Total Error Rate = (False Acceptance Rate + False Rejection Rate)/2.

resolution fabrication capturing the 3D shape and texture information of the live user's face. By contrast, replay video attacks can be easily launched simply using a smartphone to obtain a photograph or video of the target subject.

Each type of spoof attack requires a different strategy to safe guard the system. Hence an input, face video/image to a FR system should go through several modules (mixture of experts), each one focusing on detecting a single type of spoofing attack [8]. Additionally, most of the published methods on face spoof detection are based on databases (CASIA and Idiap, both released in 2012) in which the spoof videos were captured using either low resolution (USB camera) or very high-resolution (DLSR) cameras [5, 13]. The CASIA and Idiap databases did not consider mobile phone unlock scenario.

The contributions of this paper are as follows:

- Collection of a new face spoof database[3] to replicate the scenario of smartphones unlock (Nexus 5) by replaying face videos on a MacBook laptop.
- Use of moiré patterns for detecting replay attacks.
- Performance evaluation using different color channels.
- State of the art spoof detection performance for cross-database testing scenarios.[4]

2. Related Work

2.1. Literature Review

We provide a short summary of published spoof detection methods and give a brief analysis of their results. Over the years, a number of methods have been proposed for face spoofing detection for print attacks [2, 11, 13] and for replay-attacks [4, 6, 19]. Since our focus is on replay attacks, we briefly review the published methods by grouping them into two categories (Table 1): (i) methods based on face analysis and (ii) methods based on image quality analysis.

Face spoofing detection methods based on face analysis extract face-specific characteristics (physiological or behavioral) such as eye blink [13], lip or head movement [4], texture [11], and 3D shape [2, 6, 9]. Some methods used a fusion of multiple physiological or behavioral clues to detect spoof faces [16]. Although these methods report favorable results for intra-database testing, they require accurate face and/or landmark (eye) detection. Additionally, these studies did not provide evaluations in cross-database testing scenarios, which is more representative of real applications.

Biometic spoofing detection methods based on image quality analysis have been shown to have good generalization ability to different scenarios [7]. However, studies on face spoofing detection based on image quality analysis are limited. In [7], 25 image quality measures, including 21 full-reference measures and 4 non-reference measures, were used to detect spoof faces, which were also used for fingerprint and iris spoof detection. But the authors did not show how their method generalizes to cross-database testing scenarios. On a related note, while there are a number of studies on face image quality assessment [3, 15], their utility for face spoofing detection have not been explored.

2.2. Replay Attack Spoof Databases

In this section, we discuss two well-known and commonly used public-domain face spoof databases for replay attacks as well as a database we extended upon. Additionally, we will discuss how these databases were collected and their limitations.

The Idiap REPLAY-ATTACK database,[5] consists of $1,200$ video clips of photo and video replay attacks for 50 subjects [5]. Live face videos of subjects were captured using the webcam on a MacBook. Replay attacks for each subject were captured using a Cannon PowerShot SX 150 IS camera that records 720p video clips. The high-resolution camera captured replay attacks displayed on an iPhone 3GS (480×320 resolution) and iPad 1 (1024×768 resolution).

[3]Portions of the RAFS database (where subjects have given approval) will be made available to interested researchers.

[4]Cross-database testing involves, training on database A and testing on a database B, collected in a different setting from database A and with different subjects. This is in contrast to the easier, but, not realistic protocol of intra-database testing where cross-validation is used on a specific database, say A.

[5]www.idiap.ch/dataset/replayattack

Table 2. A summary of public-domain replay attack spoof databases.

Database†	# Subs.	# Videos (Live, spoof)	Live face acq. device	Spoof medium	Spoof acq. device	Subject race
Idiap REPLAY-ATTACK [5]	50	(200, 1000)	MacBook webcam (320 × 240)	iPad 1 (1024 × 768) iPhone 3GS (480 × 320)	Cannon PowerShot SX 150 IS (1280 × 720)	Caucasian 76%, Asian 22%, African 2%
CASIA [19]	50	(200, 450)	Sony NEX-5 (1280x720) USB camera (640x480)	iPad 1 (1024x768)	Sony NEX-5 (1280x720) USB camera (640x480)	Asian 100%
RAFS (this paper)	55	(55, 110)	Nexus 5 (frontal, 720 × 480)	MacBook (1280 × 800)	iPhone6 (rear: 1920 × 1080) Nexus5 (rear: 1920 × 1080)	Caucasian 44%, Asian 53%, African 3%

† We also generated 100 spoof videos for each of the Idiap and CASIA databases.

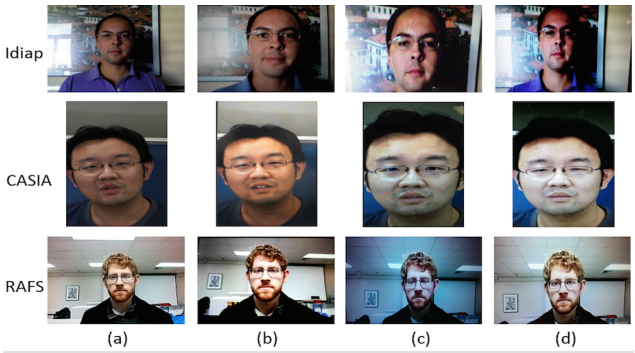


Figure 2. Sample images of live and spoof faces from Idiap (top), CASIA (middle) and RAFS (bottom) databases. (a) Live faces; (b) Original spoof faces; (c) Spoof faces generated by Google Nexus 5 using a MacBook for replay; (d) Spoof faces generated by iPhone 6 using a MacBook for replay.

The CASIA Face AntiSpoofing Database,[6] consists of 600 video clips of 50 subjects [19]. Out of the 600 video clips, 150 clips represent video replay attacks. Compared to the Idiap database, the CASIA DB used a variety of cameras (Sony NEX-5-HD, two low quality USB) to capture replay attacks displayed on an iPad.

A key drawback of the Idiap and CASIA databases is that they capture replay video attacks using either low-quality cameras or DSLR cameras that are expensive. In real world scenarios, many devices that are equipped with FR systems such as smartphones can capture replay attacks using their built-in cameras instead of an external camera. DSLR cameras with advanced features and hardware such as 35mm full frame sensor size cannot accurately portray the video quality of a smartphone's average 8.67 mm frame sensor.[7] To the best of our knowledge, no public domain database is available where the replay attack videos are captured using smartphones. To study the effects of using such videos for spoof attacks, we have extended upon the Michigan State University Mobile Face Spoofing Database (MSU MFSD) to better represent replay attacks against smartphones [17].

The smartphone face spoofing database collected in our lab called RAFS (Replay-Attack for Smartphones), contains 165 videos from 55 subjects. RAFS extends the MSU MFSD by capturing replay attacks using smartphones. Of these 165 videos, 55 videos are live face videos from the MSU MFSD that are captured using the front facing camera on a Google Nexus 5 in a controlled background environment. The remaining 110 (2 × 55) videos are spoof face videos which are captured by showing the live face videos on a MacBook screen (1280×800), and recapturing the face videos using the built-in rear camera of Google Nexus 5 and built-in rear camera of iPhone 6, respectively (see Fig. 3). At the time of this writing, Google Nexus 5 and iPhone 6 were state of the art models.

Figure 3. Demonstration of how replay attack videos are collected, using a laptop screen as the spoof medium and a smartphone as a acquisition device. This simulates how a user may launch an attack against a FR system by using a video/image found online.

In addition to the smartphone face spoofing database that was collected in our lab, we have also used 100 additional subjects from the CASIA and Idiap databases. For each subject in these two databases, one live face video is displayed on the MacBook screen, and two spoof videos are recorded using the same two smartphone cameras (Google Nexus 5 and iPhone 6) as used to capture the RAFS spoof database[8]. Therefore, we have 50 live face videos and 100 spoof face videos for both the Idiap REPLAY-ATTACK and CASIA databases. This allows us to evaluate the proposed face spoofing detection approach in cross-database scenarios.

Both the Google Nexus 5 and iPhone 6 are used to capture HD 1080p video at 30fps of live client videos being replayed on a MacBook screen (1280 × 800) to generate video replay attacks. The average standoff of the smartphone camera from the screen of the MacBook was 15 cm. This 15 cm standoff distance assured that replay videos did not contain the bezels (edges) of the MacBook screen. The

[6] www.cbsr.ia.ac.cn/english/FaceAntiSpoofDatabases.asp

[7] www.techspot.com/guides/850-smartphone-camera-hardware/page4.html

[8] Videos were not deliberately captured to include moiré patterns, most videos were captured using only a single attempt

(a) Overlay of two patterns (b) Printing (c) Screen shooting

Figure 4. Examples of moiré patterns in natural images: (a) an overlay of two patterns generates moiré patterns, (b) moiré patterns exist in color printing with halftoning, and (c) moiré patterns appear while capturing the screen of digital devices.

average duration of the replay attack video in the database is 4 seconds.[9] A major desirable property of capturing spoof videos with smartphone devices, is that it simulates input videos that may be presented to devices that contain FR systems, such as the Google Nexus 5. These face spoofing databases are summarized in Table 2. Example live and spoof face video frames are shown in Fig. 2.

For collecting the database, we did not use the front facing cameras on the Google Nexus 5 and iPhone 6 as they both lack autofocus capabilities and hence tend to provide blurry replay attack videos. In future evolutions of the smartphone, capabilities of the front facing camera will rival that of rear camera. For example, a phone developed by HTC, called Desire Eye[10], contains identical rear and front cameras (13-megapixel) and both include autofocus, a feature necessary to capture moiré pattern. Other devices encompass front facing cameras with autofocus as well. Such as the 8-megapixel frontal camera of the IQXA developed by i-mobile[11]. However at the time of this study, these devices were not released to the public.

Many DSLR cameras come equipped with anti-aliasing filters that sit immediately above the photo sensor (CCD array in most cameras) to reduce the occurrence of moiré patterns.[12] These filters reduce the sharpness of an image by smoothing the transitions between pixels, in turn reducing moiré patterns (but not completely eliminating them). Low quality webcams often lack autofocus capability or have relatively slow autofocus speed. Because of these reasons, webcams often capture blurry images when shooting a digital screen, and will not produce sharp images, showing little or no moiré patterns. These two types of cameras also do not replicate the real application scenarios of interest, namely user authentication on smartphone.[13]

[9] We did not record spoof videos with longer durations because a face spoofing detection system is expected to provide a quick response. (e.g., less than 1 sec.)

[10] www.htc.com/us/smartphones/htc-desire-eye/

[11] www.malaysianwireless.com/2013/08/i-mobile-iq-xa-malaysia/

[12] www.lifepixel.com/blog/anti-aliasing-low-pass-filter-removal

[13] Smartphone users worldwide will total 1.75 billion in 2014: www.emarketer.com/Article/Smartphone-Users-Worldwide-Will-Total-175-Billion-2014/1010536

3. Moiré Pattern Analysis

3.1. Moiré Pattern Aliasing

Moiré patterns are an undesired aliasing of images produced during various image display and image acquisition processes [1]. Aliasing refers to an effect in which reconstructed signals do not well represent the original signal. Moiré patterns appear when two or more patterns are overlaid on top of each other, resulting in a third new pattern (Fig. 4 (a)).[14] In color printing with CMYK (cyan, yellow, magenta, and black) halftoning model, moiré patterns are often inevitable (Fig. 4 (b)).[15] Moiré patterns are also observed in the screen shooting photography (Fig. 4 (c)).[16]

The use of aliasing and the appearance of moiré patterns go hand and hand. Images with moiré pattern do not accurately represent real world scenes. Cameras that capture sharp images of a digital screen not only have more information, but also capture the information more precisely, leading to moiré patterns. The fundamental reason for moiré patterns in screen shooting photography is because of the spatial frequency differences between the display and the acquisition devices. For example, when the scene (on the display of a replay device) contains repetitive details that exceed the resolution of a camera, moiré patterns are observed.

The display of digital devices (laptops, mobile devices, and tablets) exhibit a naturally occurring fixed repetitive pattern created by the geometry of color elements that are used for color displays. Therefore, whenever a video of a digital screen is recorded, moiré patterns will naturally present themselves. Analyzing the 310 replay attack videos that we have generated from the three databases (Idiap, CASIA and RAFS), a distinct moiré pattern can be recognized across the replay attack video frames (see Fig. 5).

3.2. Moiré Pattern Representation

By comparing the spoof face videos and the live face videos, we find that moiré patterns often exist in the entire spoof video frame, which appear as a distinct texture pattern overlaid on a live video frame. This inspired us to capture moiré patterns using a number of well known texture descriptors, such as MLBP [12] and SIFT [10] to use for spoof detection.

We first decode each video into individual frames using the FFmpeg library.[17] Given an input frame (can also be the detected face or a face region), it is first divided into 32×32 patches with an overlap of 16 pixels between every two

[14] www.ishootshows.com/2012/04/09/understanding-moire-patterns-in-digital-photography/

[15] users.ecs.soton.ac.uk/km/imaging/course/moire.html

[16] blog.ishback.com/?cat=132

[17] www.ffmpeg.org

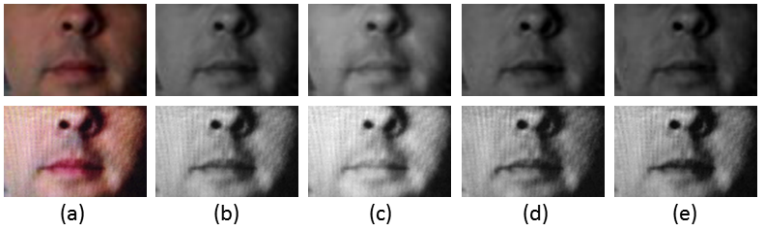


Figure 5. Examples of live video frames (top row) and spoof video frames we collected (bottom row) for one subject from the Idiap database. Video frames are shown using the (a) RGB image, (b) grayscale image, (c) red channel, (d) green channel, and (e) blue channel, respectively. To show the moiré patterns clearly, we magnify the bottom portion of a face (below the nose), however moiré patterns exist in the entire spoof video frames as well.

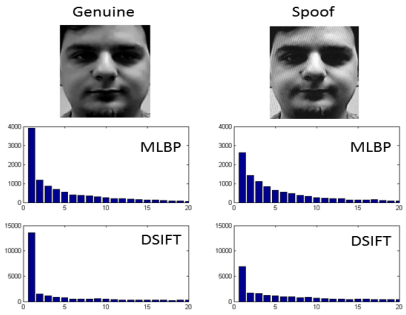


Figure 6. Examples of multi-scale LBP (MLBP) and densely sampled SIFT (DSIFT) features extracted from (a) a frame of live face, and (b) a frame of spoof face. The differences in histogram distribution allow us to differentiate a live face from a spoof face. The horizontal and vertical axes are histogram bins and bin frequency counts, respectively.

successive patches. For each image patch, we use multi-scale LBP (MLBP) to capture the characteristics of moiré patterns. The MLBP features are calculated as

$$f_{MLBP}(I) = \{LBP_{P,R}\}_{(P,R)\in\{(8,1),(24,3),(40,5)}\,, \quad (1)$$

where R and P define the individual scales (radii) and numbers of sampling points, respectively; $LBP_{P,R}$ follows the standard definition of a single scale LBP [12]

$$LBP_{P,R} = \sum_{p=0}^{P-1} sign\left(I(p) - I(c)\right), \quad (2)$$

where $I(c)$ and $I(p)$ are the intensities of the current pixel c and the sampling point p, respectively. The MLBP features from individual patches are concatenated together to construct a histogram.

To show the robustness of the proposed approach against different texture descriptors, we also used densely sampled SIFT (DSIFT) features in our experiments. The DSIFT features from each image patch are calculated using 8 orientation bins and 16 segments. Examples of MLBP and DSIFT histograms from a frame of live face, and a frame of spoof face are shown in Fig. 6.

In most FR literature, the MLBP and SIFT features are usually extracted from the grayscale (intensity) images.

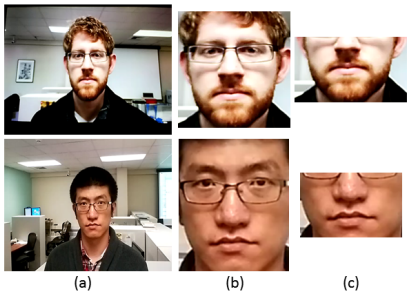


Figure 7. Examples of three different image regions that are used for moiré pattern analysis: (a) the whole video frame, (b) the detected face image, and (c) the bottom half of the face image.

Table 3. Dimensionality of feature vectors using different image regions and descriptors.

	Whole Frame	Whole Face	Bottom Face
MLBP	62,776	11,328	3,540
SIFT	191,808	34,560	9,600
Fusion	254,584	45,888	13,140

However, for the face spoof detection using moiré patterns, we observe that, the moiré patterns in one of the channels (red, green and blue) of an input image can be more discriminative than the other two channels or the intensity image (see Figs. 5 (b-e)). The possible reason is that some of the color channels may not retain the facial texture details very well, accentuating the moiré patterns with higher contrast for spoof video frames.

As pointed out earlier, moiré patterns exist not only in the facial region but also in the whole video frame containing the face. This make it possible to detect spoof face without first performing face detection operation. This can be very useful for face spoof detection under less-cooperative scenarios (non-frontal face), where face detection may be challenging. In the experimental section, we show the robustness of the proposed face spoof detection method by using the (i) whole video frame, (ii) detected face image, and (iii) bottom part of the face image. Examples of the three image regions are shown in Fig. 7. The feature vectors used in our experiments include MLBP, DSIFT, and the concatenation of MLBP and DSIFT extracted from three different regions. Individual histogram bins are used as features.

3.3. Multi-frame Based Classification with Voting

Given a texture feature vector (Table 3), we train a SVM classifier with a RBF kernel (using optimized parameters) as a live or spoof classifier.[18] In order to classify a video as live or as a spoof, we utilize multiple frames in the video. The SVM classifier outputs a confidence score for each video frame (live or spoof). Therefore, we keep track of how many frames in each video are labeled as live or as a spoof. In essence, the class that has more than 50% of the votes determines the class of the video.[19]

4. Experimental Results

4.1. Testing Protocols

We evaluate the proposed approach under both cross-database and intra-databases testing scenarios. It is now generally accepted that intra-database testing (training and test images/video, while distinct, were captured in the same environment and possibly of the same subjects) does not represent real world scenarios, as they lack generalization ability [14]. We use the Idiap, CASIA and RAFS (collected in our lab) databases to conduct the cross-database testing (training and test sets contain different subjects and were captured in different locations and imaging environments). For the testing on each of the three databases, the other two databases are used to train the proposed moiré patterns based approach.

We also conduct intra-database experiments on each of the three databases using the following two protocols: (i) 5-fold cross validation, and (ii) the protocols provided with the Idiap and CASIA databases, so that we can provide comparisons with the published methods.

4.2. Cross-database Testing

- **Influence of the number of frames.** We evaluated the performance of the proposed approach by using the first 1, 5, 10 and 20 frames of each (live and spoof) video. A concatenation of the MLBP and DSIFT features extracted from the intensity face image was used. Figure 8 shows that the proposed approach achieves relatively better performance with the first 5 frames. Due to continuous autofocus in smartphones, moiré patterns present themselves in frames with sharp focus. In practice, a live vs. spoof decision must be made quickly and reliably. Using the first 5 frames (< 0.2 sec.) is a good tradeoff between accuracy and speed.

- **Influence of different color channels.** We analyzed the grayscale, and red, green and blue channels of the detected face image using 5 frames for voting and a concatenation of MLBP and DSIFT features. Table 4 shows that different color channels capture different amount of texture to represent moiré patterns. Red channel gives better results because it has a higher contrast between the moiré pattern and the facial texture.

- **Influence of different image regions.** We study the effect of different image regions (whole video frame, detected face image, and bottom half of a face), from where moiré pattern features are extracted. Again, the concatenation of MLBP and DSIFT features are used, and 5 frames are used for each live and spoof video. Table 5 shows that the detected face image and the bottom half of the face lead to the same average performance (76.1%) on the three databases. The bottom half of the face minimizes hair style variations among different subjects, and also provides a lower dimensional feature vector.

- **Influence of different descriptors.** We use three different features (MLBP, DSIFT and the concatenation of MLBP and DSIFT) to represent moiré patterns. Figure 9 shows that using any one of these descriptors results in state of the art performance for cross-database testing. However, the MLBP descriptor gives similar performance as the concatenation of MLBP and DSIFT, but MLBP has a lower dimensional feature vector.

- **Overall accuracy.** We now report the accuracy of the proposed approach on the three databases by using MLBP features to represent moiré pattern from the red channel of the bottom part of the face using

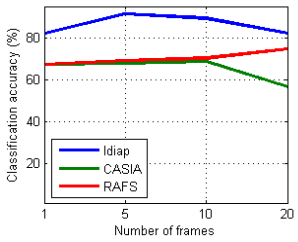


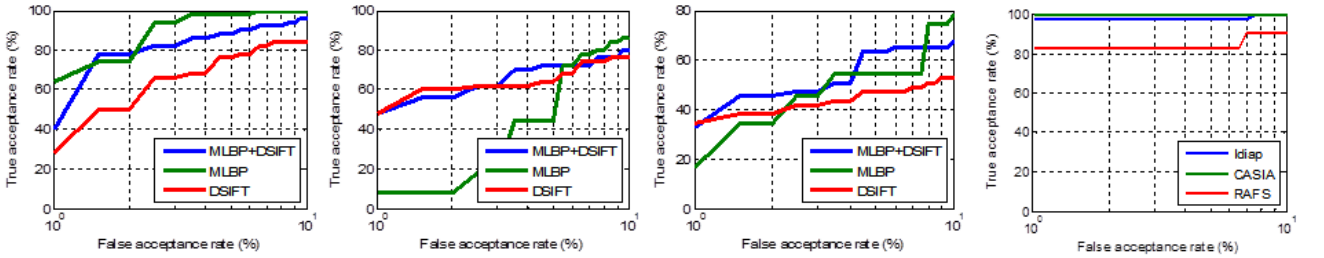
Figure 8. Number of frames vs. the performance of moiré pattern based approach for replay attack detection under cross-database testing. A concatenation of the MLBP and DSIFT features from the detected face image is used.

Table 4. Color channels (grayscale, red, green and blue) vs. performance of the moiré pattern based approach under cross-database testing. One database is used for testing on a classifier trained on the other two databases.

Database	Grayscale	Red	Green	Blue
Idiap	91.3%	96.7%	91.3%	86.0%
CASIA	68.0%	68.0%	68.0%	67.3%
RAFS	69.1%	76.4%	69.7%	80.0%

[18] LIBSVM is used: `www.csie.ntu.edu.tw/~cjlin/libsvm`

[19] We also tried the score level fusion of all the frames, but it gives worse performance than the proposed voting scheme.



(a) Cross-DB test on Idiap (b) Cross-DB test on CASIA (c) Cross-DB test on RAFS (d) Intra-DB test on Idiap, CASIA and RAFS

Figure 9. Performance of the moiré pattern based approach under cross-database and intra-database testing. (a-c) shows the robustness of the proposed approach to different features (MLBP, DSIFT, and feature-level fusion of MLBP and DSIFT) on the Idiap, CASIA, and RAFS databases. (d) performance of the proposed approach on the three databases under intra-database testing.

Table 5. Classification accuracy of moiré patterns from different regions (whole frame, detected face, and bottom part of face) on the Idiap, CASIA and RAFS databases under cross-database testing. Results are based on a concatenation of the MLBP and DSIFT.

Database†	Whole frame	Whole face	Bottom face
Idiap	72.7%	91.3%	87.3%
CASIA	47.5%	68.0%	70.7%
RAFS	72.0%	69.1%	70.3%

†One of the three databases (shown in the first column) is used for testing while the other two databases are used for training.

the first 5 frames for voting. Classification accuracies of 88.0%, 67.3% and 85.5% are obtained on the Idiap, CASIA and RAFS database, respectively, under cross-database testing scenarios. The performance on the CASIA database is not as good as the other two databases because the live face videos in the CASIA database have a much higher resolution than the Idiap and RAFS databases.

Examples of correct and incorrect classifications are shown in Fig. 10. The examples in Fig. 10(c) are misclassified because the subjects have beards which MLBP may classify as having the same texture as moiré patterns. Subjects in Fig. 10(d) are misclassified due to blurry frames caused by the camera's inability to autofocus in time to capture sharp images.

The best reported performance on the Idiap and CASIA databases under cross-database testing scenarios have HTERs of 47.1% and 48.3%, respectively [14]. The proposed approach achieves 18.0%, 49.0% and 11.4% HTERs on the Idiap, CASIA and RAFS, respectively, under cross-database testing scenarios. While the proposed approach achieves similar performance to [14] on the CASIA database, our method significantly outperforms [14] on the Idiap database, which shows a better generalization ability of the proposed moiré pattern based method.

In terms of the computational cost of the proposed approach, MBLP feature extraction takes 0.09 seconds per frame, and classification takes 0.02 seconds per frame, which results in a decision for an input face video in 0.47 seconds using the first 5 frames. All the times are profiled with a Matlab implementation on a Windows 7 platform with Intel Core 2 quad 3.0 GHz CPU and 8GB RAM.

4.3. Intra-database Testing

We also evaluate the proposed approach under the intra-database testing scenarios on the Idiap, CASIA, and RAFS databases, using 1 live video and 2 spoof videos (captured by Google Nexus 5 and iPhone 6) for each subject. Table 1 and Figure 9(d) show that the proposed approach achieves 3.3%, 0.0%, and 11.3% HETRs on the Idiap, CASIA and RAFS database, respectively. On the Idiap database, our approach (3.3%) gives slightly larger HTER than the state of the art method (1.3%) [4], but no cross-database testing result was reported in [4]. On the CASIA database, our approach (0.0%) achieves much smaller HTER than the state of the art (11.8%) [18]. Again, no cross-database testing result was reported in [18].

5. Summary and Conclusions

Spoofing attacks are a menace to biometric systems in terms of public perception and adoption. Face recognition systems can be easily targeted due to the low cost in launching replay video attacks. We have proposed a robust replay attack detection method for FR systems that can generalize well, especially for cross-database testing which portrays real world scenarios. We analyze the moiré pattern aliasing that is observed during recapture of video or photo replays on a digital screen. In order to analyze this phenomenon, we collected a database, called RAFS, that contain replay video attacks towards smartphones. The moiré patterns can be detected using MLBP and DSIFT features. Evaluations for intra-database test show that the proposed methods returns state of the art accuracies in detecting replay video attacks, however intra-database results do not portray real world scenarios. Cross-database results show that the proposed method generalizes well in detecting replay video at-

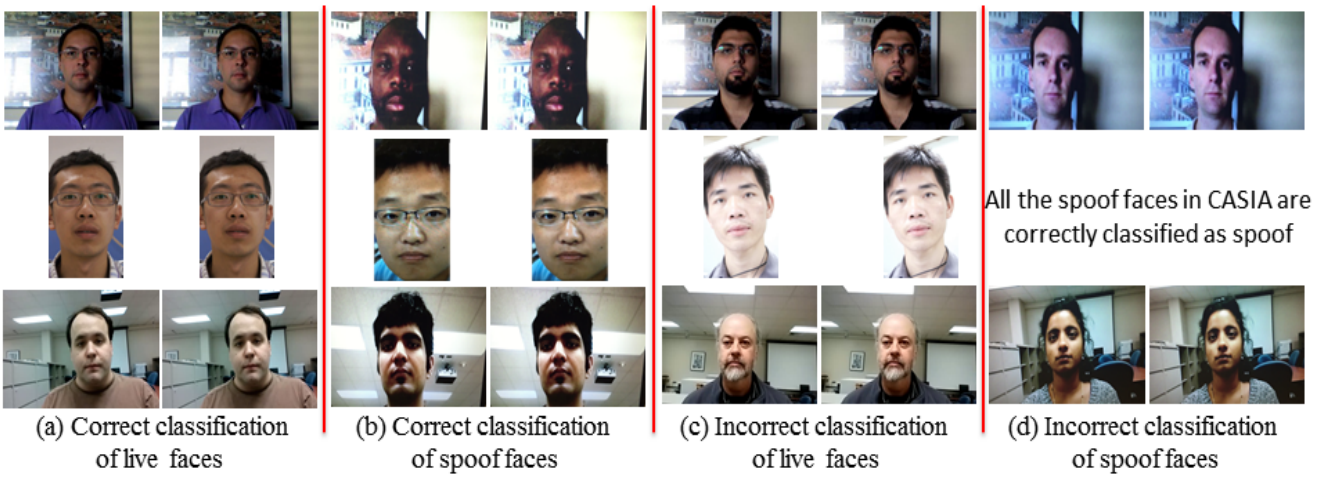


Figure 10. Examples of correct and incorrect classifications by the proposed approach in cross-database testing on the Idiap (top row), CASIA (middle row), and RAFS (bottom row) databases. Correct classifications for (a) live faces and (b) spoof faces, and incorrect classifications for (c) live faces and (d) spoof faces.

tacks compared to published methods.

For future work, we plan to extend the moiré pattern based method to detect replay photo attacks. Additionally, we will generate new replay video attacks using a smartphone that contains autofocus ability on its front facing camera and expand our experiments to include a variety of spoofing mediums, such as smartphones and tablets.

6. Acknowledgement

This research was supported by CITeR grant (14S-04W-12).

References

[1] I. Amidror. *The Theory of the Moiré Phenomenon Volume I: Periodic Layers, 2nd ed.* Springer, 2009.

[2] W. Bao, H. Li, N. Li, and W. Jiang. A liveness detection method for face recognition based on optical flow field. In *Proc. IASP*, pages 233–236, 2009.

[3] L. Best-Rowden, H. Han, C. Otto, B. Klare, and A. K. Jain. Unconstrained face recognition: Identifying a person of interest from a media collection. *IEEE Trans. Inf. Forensics Security*, 9(12):2144–2157, Dec. 2014.

[4] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh. Computationally efficient face spoofing detection with motion magnification. In *Proc. CVPR Workshops*, pages 105–110, 2013.

[5] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Proc. IEEE BIOSIG*, pages 1–7, 2012.

[6] M. De Marsico, M. Nappi, D. Riccio, and J. Dugelay. Moving face spoofing detection via 3d projective invariants. In *Proc. ICB*, pages 73–78, March 2012.

[7] J. Galbally, S. Marcel, and J. Fierrez. Image quality assessment for fake biometric detection: Application to iris, fingerprint and face recognition. *IEEE Trans. Image Process.*, 23(2):710–724, Feb 2014.

[8] J. Komulainen, A. Hadid, M. PietikÌainen, A. Anjos, and S. Marcel. Complementary countermeasures for detecting scenic face spoofing attacks. In *ICB*, 2013.

[9] J. Li, Y. Wang, T. Tan, and A. K. Jain. Live face detection based on the analysis of fourier spectra. In *Proc. SPIE*, pages 296–303, 2004.

[10] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, pages 1150–1157, 1999.

[11] J. Määtta, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Proc. IJCB*, pages 1–7, 2011.

[12] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):971–987, Jul 2002.

[13] G. Pan, L. Sun, Z. Wu, and S. Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *Proc. ICCV*, pages 1–8, Oct. 2007.

[14] T. F. Pereira, A. Anjos, J. De Martino, and S. Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *Proc. ICB*, pages 1–8, June 2013.

[15] Y. Sun, M. Tistarelli, and D. Maltoni. Structural similarity based image quality map for face recognition across plastic surgery. In *Proc. BTAS*, pages 1–8, Sept 2013.

[16] R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori, and F. Roli. Fusion of multiple clues for photo-attack detection in face recognition systems. In *Proc. IJCB*, pages 1–6, Oct 2011.

[17] D. Wen, A. K. Jain, and H. Han. Face spoof detection with image distortion analysis. In *IEEE Trans. Information Forensic and Security*, 2015.

[18] J. Yang, Z. Lei, S. Liao, and S. Li. Face liveness detection with component dependent descriptor. In *Proc. ICB*, pages 1–6, June 2013.

[19] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *Proc. ICB*, pages 26–31, 2012.