Face/Fingerphoto Spoof Detection under Noisy Conditions by using Deep Convolutional Neural Network

Masakazu Fujio[1], Yosuke Kaga[1], Takao Murakami[2], Tetsushi Ohki[3] and Kenta Takahashi[1]
[1]*Security Research Dept., Hitachi, Ltd., Tokyo, Japan*
[2]*Advanced Cryptosystems Research Group,*
National Institute of Advanced Industrial Science and Technology, Tokyo, Japan
[3]*Department of Computing, Shizuoka University, Shizuoka City, Japan*

Keywords: Biometrics, Spoofing, LBP, CNN, Deep Learning, Mobile, Blurriness.

Abstract: Most of the generic camera based biometrics systems, such as face recognition systems, are vulnerable to print/photo attacks. Spoof detection, which is to discriminate between live biometric information and attacks, has received increasing attentions recently. However, almost all the previous studies have not concerned the influence of the image distortion caused by the camera defocus or hand movements during image capturing. In this research, we first investigate local texture based anti-spoofing methods including existing popular methods (but changing some of the parameters) by using publicly available spoofed face/finger photo/video databases. Secondly, we investigate the spoof detection under the camera defocus or hand movements during image capturing. To simulate image distortion caused by camera defocus or hand movements, we create blurred test images by applying image filters (Gaussian blur or motion blur filters) to the test datasets. Our experimental results demonstrate that modifications of the existing methods (LBP, LPQ, DCNN) or the parameter tuning can achieve less than 1/10 of HTER（half total error rate）compared to the existing results. Among the investigated methods, the DCNN (AlexNet) can achieve the stable accuracy under the increasing intensity of the blurring noises.

1 INTRODUCTION

With the exponential growth of the smartphone market, financial services are also accelerated by the development of various services on mobile devices, such as mobile payments, money transfer, and all banking related transactions.

As the mobile e-commerce continues to grow, the biometrics authentications are attracting more attentions for the secure and easy-to-use authentication methods, as the alternative for the insecure and inconvenient Password/PIN authentication methods.

Biometrics can implement the convenient user authentication, but on the other hand, it is vulnerable to be spoofed by the fake copies of the user biometric features made of commonly available materials such as clay and gelatines. For example, the gummy finger model attacks for the fingerprint biometrics solutions demonstrate the possibilities of the unauthorized accesses. Especially the generic camera based biometrics has the higher risk of spoofing by the printed photos and videos (preparation costs are low). For that reason, the spoof detection, which is to discriminate between live faces/fingers and attacks, has received increasing attentions recently (Keyurkumar et al., 2015; Bai et al., 2010).

Typical anti-spoofing techniques can be coarsely classified into three categories based on clues used for spoof attack detection: (i) motion analysis based method, (ii) texture analysis based methods, and (iii) image qualities analysis based methods.

(i) Motion analysis based methods

These methods, which are effective to counter printed photo attacks, capture the movement clues such as eye blinks (Gang et al., 2007) and lip movements (Avinash et al., 2014), which are very important cues for vitality. But in the case of the face biometrics, the system needs accurate detections of facial parts such as eyes, lips and so on. Furthermore,

Fujio, M., Kaga, Y., Murakami, T., Ohki, T. and Takahashi, K.
Face/Fingerphoto Spoof Detection under Noisy Conditions by using Deep Convolutional Neural Network.
DOI: 10.5220/0006597500540062
In *Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2018) - Volume 4: BIOSIGNALS*, pages 54-62
ISBN: 978-989-758-279-0

simply capturing movement clues are not enough for the presentation attacks by videos.

(ii) Texture analysis based methods

These methods capture the texture features appeared on natural scenes, photo papers, and displays under the assumption that surface properties of real faces and prints are different (Diego and Giovanni, 2015; Tiago, 2012, 2014; Ivana, 2012; Juho et al., 2012).

Texture based methods such as Local Binary Patterns (LBP) (Matti et al., 2011) have achieved significant success on the Idiap and CASIA databases (Ivana et al., 2012; Zhiwei et al., 2012). For example, The Half Total Error Rate (HTER) on the Idiap database was reduced from 13.87% in (Ivana et al., 2012) to 6.62% in (Samarth, 2013). Unlike motion based methods, texture based methods need only a single image to detect a spoofing.

Other types of texture analysis methods adopt the frequency domain features (Jiangwei et al., 2004). For example, low resolution printed images have a high-frequency spectral magnitude in the frequency domain, caused by periodic dot printing (Xiaofu et al., 2009). Jiangwei et al. (2004) described a method for detecting print-attack face spoofing by exploiting differences in the 2D Fourier spectra of live and spoofed images. The method assumes that photographs are normally smaller in size and contain less high-frequency components compared to real faces. Then their method likely fails for higher-quality samples.

(iii) Image qualities analysis based methods

These methods capture the degradation of the image qualities caused by presenting photographs or videos to the generic cameras (Javier et al., 2014; Diogo and Ricardo, 2015; Di et al., 2015). For example, printed/displayed images usually have lower resolution, narrow dynamic range, specular reflection, reduced image contrast and defocused blurriness. But those image quality degradations also appear in both genuine and spoofed face images, it is not simple to distinguish that the image distortions are caused by spoofing or camera operations.

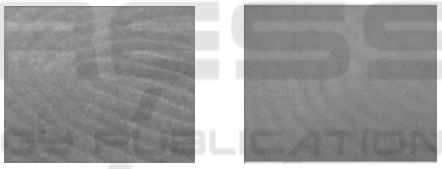
The problems of the almost all the existing studies are that they have not concerned the influence of the image distortion caused by the camera defocus or hand movements during an image capturing.

In this research, we first investigate local texture based anti-spoofing methods including existing popular methods (but changing some of the parameters) by using publicly available spoofed face/fingerphoto/video databases (Replay-Attack Database and Spoofed Fingerphoto Database). Secondly, we investigate the spoof detection under the camera defocus or hand movements during image capturing. To simulate image distortion caused by camera defocus or hand movements, we create blurred images by applying image filters (Gaussian blur or motion blur filters) to the test datasets. For the training images, we do not apply image filters.

The remaining of the paper is organized as follows. Section 2 contains the examined schemes of the anti-spoof technique and provides an explanation of the countermeasures to photo/display attacks in fingerphoto or face recognition. Section 3 details experimental protocols, the dataset statistics, parameters used in the algorithm and the results obtained. Section 4 concludes the paper.

2 PROPOSED METHODS

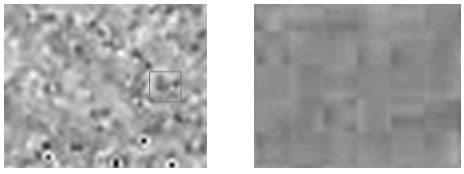
To design the anti-spoofing countermeasures, we investigated fingerphoto spoofed image database (Archit et al., 2016). Fig. 1 shows magnified images (The left side is a genuine image, and the right side is a spoofed image.). From the Fig. 1, we can see block noises in the spoofed images.



(a) genuine (b) spoofed

Figure 1: Magnified fingerphoto images.

To highlights the noises in the images, we performed image enhancement of the above images based on wavelet transform (Fig. 2). From the Fig. 2, we can see repeated block noise artifacts in the spoofed images (The square frame in the left image in the fig. 2 shows a block of 8x8 pixels).



(a) genuine (b) spoofed

Figure 2: Wavelet transformed images (left：genuine, right：spoofed).

To capture the noise features found in the preliminary analysis, we focus on the LBP (Matti et al., 2011) and the block noise indicator (Zhou et al.,

2000, 2002). The block noise indicator is used to quantify the magnitude of block artifacts caused by the application of lossy compression algorithms such as JPEG compression. And used for the judgment of the image compression algorithms (Zhou et al., 2000, 2002). In the rest of this section, we will explain anti-spoofing techniques based on handcrafted features or the automatic feature extraction based on CNN (convolutional neural network).

2.1 SVM with Handcrafted Features

We used the following 3 handcrafted feature vectors to train Support Vector Machine (SVM) classifiers.

(1) WLBP (Wavelet transformed Local Binary Pattern)

LBP (Local Binary Patterns) is the 8-bit encoding based on the comparisons of the magnitudes of the luminance between the focused pixel and neighboring 8 pixels. LBP is widely used for the image classification tasks in the literature.

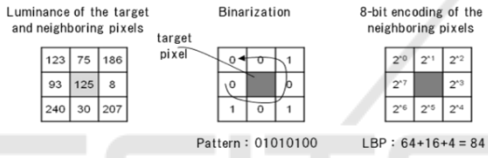


Figure 3: LBP features of the each pixel.

In addition to the LBP of original image size, we extracted LBP features from compressed images and contrast enhanced images. All features are concatenated and used for the training of the linear SVM.

(a) Calculation of LBP from the original images:

(b) Image enhancement (wavelet transformation):

To enhance the noise patterns in the images, perform wavelet transformations and calculate LBP of the transformed images

(c) Image compression:

Base on the preliminary study about block noise pattern, we set the compression ratio as 3/8 ("3" is the kernel size of LBP, and "8" is the size of the observed blockiness).

(d) Feature fusion:

Concatenate the three LBP (original, wavelet transformed, compressed), and use for the training of the linear SVM

(2) NRPQA (No-Reference Perceptual Quality Assessment)

This measure applies the block artifact indicator (2002) for the ant-spoof detections. Zhou et al. (2002) describes perceptual quality assessment of JPEG compressed images by calculating three measures, inter block differential, intra block differential and zero-crossing rate.

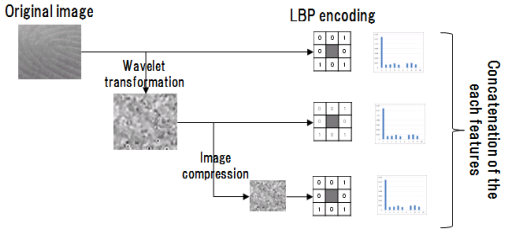


Figure 4: 3 types of LBP features.

We denote the test image signal as x (m; n) for $m \in [1, M]$ and $n \in [1, N]$, and calculate a differential signal along each horizontal line:

$$d_h(m,n) = x(m,n+1) - x(m,n), \quad n \in [1, N-1] \tag{1}$$

The features are calculated horizontally and then vertically. The amount of blockiness is estimated as the average differences across boundaries.

$$B_h = \frac{1}{M(\lfloor N/8 \rfloor - 1)} \sum_{i=1}^{M} \sum_{j=1}^{\lfloor N/8 \rfloor - 1} |d_h(i, 8j)| \tag{2}$$

Second, we estimate the activity of the image signal. The second activity measure, A_h is the average absolute difference between in-block image pixels.

$$A_h = \frac{1}{7}\left[\frac{1}{M(N-1)} \sum_{i=1}^{M} \sum_{j=1}^{N-1} |d_h(i,j)| - B_h \right] \tag{3}$$

The third activity measure Z_h is the zero-crossing (ZC) rate. Horizontal Zero-Crossing (ZC) means that there is a change of the sign of the value $d_h(m,n)$ between n and n+1 ($n \in [1, N-2]$):

$$z_h(m,n) = \begin{cases} 1 & horizontal\ \ ZC\ \ at\ \ d_h(m,n) \\ 0 & otherwise \end{cases} \tag{4}$$

ZC is then estimated by using $Z_h(m,n)$ as follows:

$$Z_h = \frac{1}{M(N-2)} \sum_{i=1}^{M} \sum_{j=1}^{N-2} |z_h(m,n)| \tag{5}$$

For the detail of these measures, please refer to (Zhou et al., 2002). In this study, we trained linear-SVM with these 6 features $\{ B_h A_h, Z_h, B_h A_h, Z_h \}$.

(3) LPQ (Local Phase Quantization)

The local phase quantization (LPQ) (Timo, 2008) is a method based on the blur invariance property of the Fourier phase spectrum and uses the local phase information extracted using 2-D discrete Fourier transform or short term Fourier transform (STFT), computed over a rectangular region. The STFT over

a region of the N by N neighborhood N_x of image g (x) with each position of the pixel x is defined by

$$H(u,x)=\sum_{y\in N_x} h(x-y)e^{-j2\pi\ uTy} \quad = w_u{}^T h_x \tag{6}$$

where w_u is the basis vector of the 2D discrete Fourier transform at a frequency u while h_x stands for the vector containing all N^2 pixels.

In the task of spoofing detection, we expect that LPQ is tolerant for the distorted images caused by defocused images or motion blurred images, which are commonly seen in both printed photos, replayed video attacks and real faces/fingers.

2.2 Deep Convolutional Neural Network (DCNN)

Convolutional Neural Networks (Alex et al., 2012) have demonstrated state-of the-art performances in a variety of image recognition benchmarks, such as MNIST (Yann and Corinna., 2010), CIFAR-10, CIFAR-100 (Alex and Geoffrey, 2009), and ImageNet (Alex et al., 2012).

In this study, we used AlexNet (Alex et al., 2012). This model won both classification and localization tasks in the ILSVRC-2012 competition. This model also exhibited good results on the spoof detection on the Replay-Attack Database (HTER<0.5%) (Koichi et al., 2017).

- CNN Model : AlexNet without pre-trained weights.

As the preliminary experiments, we tried various sizes of the image compressions and cropping sizes for the training of the DCNN (AlexNet) models. Based on the results of the preliminary experiments, we choose the combination of image resizing:256 pixels and image cropping size:227 pixels, which exhibited the highest accuracy.

We set the parameters for the training of CNN as follows:

Output layer number: 2, Resized image size: 256, random cropping size: 227, batch size: 100, epoch size: 300,000, learning rate: 0.01, weight decay: 0.004.

In the above settings, we do not use image augmentation (except for the random image cropping). We consider that not only foreground regions but also of images (such as faces and finger) but also both background images and foreground images have discriminative cues for the spoof detection.

3 EXPERIMENTS

In this section, we first provide an overview of the datasets in our experiments, and present our initial results for the proposed methods in previous section (one is linear-SVM with the handcrafted feature vectors, and the other is Deep Convolutional Neural Network (DCNN)).

3.1 Database Descriptions

To evaluate the spoofing detection accuracies, we used two databases, one is “Replay Attack Database (face) and the other is “Spoofed Fingerphoto Database (finger)”.

A) **Spoofed Fingerphoto DB**

Table 1: Summary of the Spoofed Fingerphoto DB.

#	Condition	Description
1	Lighting condition	2types (Indoor/Outdoor)
2	Background	2types (White/Natural)
3	Image size	Genuine image：3264x2448
4		Spoofed image：2332x1132
5	Number of images	Genuine image: 4096（64x2x8x4）
6		Spoofed image: 8192（64x2x2x8x4）

Table 2: Attack Protocols for Fingerphoto Spoofing.

#	Capture	Display
1	OnePlus One phone	iPad
2	OnePlus One phone	Laptop
3	OnePlus One phone	Nexus
4	OnePlus One phone	Printout
5	Nokia	iPad
6	Nokia	Laptop
7	Nokia	Nexus
8	Nokia	Printout

Following the setup of Tanja (2016), genuine images (4096) were split into the gallery and probe data (each has 2048 images). Then the images used for generating spoof images (512) were excluded from genuine images (1536), and spoofed images were added (4096) to the training of the each model (total 5632).

From probe data, genuine images (2048) + imposter images (4096) were used for the evaluation data set.

B) **Replay-Attack DB**

The 2D face spoofing attack database consists of 1,300 video clips of photo and video attack attempts of 50 clients, under different lighting conditions. The size of the image is 320 x240.

Table 3: Attack protocols for face spoofing.

#	Setting	Description
Capture condition	Lighting condition	2types (controlled/adverse)
Attack protocols	Display devices	5types (mobile photo/mobile video/high-resolution photo/high-resolution video/high-resolution print)
	Attack modes	2types (hand/fixed)

Training data size:

Attack Video: 300 clips (15 (clients) × 2 (lighting) × 5 (devices) × 2 (modes))

Real Video: 60 clips (15 (clients) × 2 (lighting) × 2 (shots))

Test data size:

Attack Video: 400 clips (20 (clients) × 2 (lighting) × 5 (devices) × 2 (modes))

Real Video: 80 clips (20 (clients) × 2 (lighting) × 2 (shots))

Following the setup of Ito (2017), we used the "training set" for the training, and evaluated the spoofing detection accuracies with the "test set". To train the face spoof detection model, we extracted the all frame images from the video clips, and both training and evaluations are performed for those images.

3.2 Spoof Detection Accuracy

In this experiment, we compare the spoof detection accuracy of the investigated methods with the baseline methods and state of the art methods on two databases: Replay-Attack Database and Spoofed Fingerphoto Database. Table 4 and Table 5 show the HTER (half total error rate) of the spoof detection accuracies for the each attack type, such as printing, photo, movies. For the Replay-Attack DB, we used all frames in the movie clips both for the training and testing.

Table 4 shows that the proposed DCNN method outperforms the other methods. For the Fingerphoto Spoof DB (Archit et al., 2016), spoof detection accuracies of the proposed methods (NRPQA+SVM and DCNN) exhibited less than 1/10 error rates compared to the Archit et al. (2016) 's LBP based method.

Table 5 shows that the proposed DCNN method outperforms the other methods, even better than the other state-of-the-art DCNN method (Koichi et al., 2017; Jianwei et al., 2014). The differences of the our DCNN model and the Koichi's DCNN model are the preprocessing of input images and the cropping size of the images. Our model compresses the input images (256x256) before cropping, but Ito's model does not. The cropping size of our model is (227x227), a little bit larger than Ito's model (240x180). As is the case with WLBP features, the image compression process may contribute to capturing the differences between the real /spoofed images. In the case of the Replay-Attack Database, the method NRPQA+SVM shows the low detection accuracies, especially for the "Highdef" samples, which means high-resolution photos and videos images. This may mean that the examined NRPAQ features may be specific features that is prominent to the Spoofed Fingerphoto Database. But only our DCNN model shows high accuracy (HTE < $2/10^{-6}$, by using "rule of three") for the "Highdef" images.

Table 4: Summary of the evaluation results (finger).

		Half Total Error Rate (%)					
Attack Scenario		Proposed Methods			Baseline		
Display	Capture	DCNN	WLBP+SVM	NRPQA+SVM	LBP+SVM	LPQ+SVM	LBP+SVM [Tanja2016]
Print	Nokia	**0.024**	**0.05**	**0.0**	0.66	4.71	6.05
	OPO	**0.024**	**0.02**	**0.0**	0.42	1.83	4.85
iPad	Nokia	**0.2**	**0.12**	**0.0**	2.52	3.15	3.12
	OPO	**0.024**	0.83	**0.0**	0.39	0.71	5.27
Nexus	Nokia	**0.024**	0.78	**0.0**	0.85	3.79	1.39
	OPO	**0.024**	0.32	0.24	0.56	3.13	0.24
Laptop	Nokia	**0.024**	0.20	0.24	5.76	20.8	4.48
	OPO	**0**	0.17	0.39	0.42	1.8	2.31
Total		**0.04**	0.9	0.06	3.56	4.8	3.71

Table 5: Summary of the evaluation results (face).

	Half Total Error Rate (%)					
Attack Scenario	Proposed Methods			Baselines		State-of-the-art
	DCNN(Ours)	WLBP+SVM	NRPQA+SVM	LBP+SVM	LPQ+SVM	DCNN [Itoh' 17]
Print	**0.0**	**0.01**	1.84	12.4	7.9	N/A
Mobile	**0.0**	**0.4**	1.14	2.5	25.0	N/A
Highdef	**0.0**	7.8	17.2	12.7	32.2	N/A
Total	**0.01**	4.98	11.1	16.5	30.2	0.52

To investigate that which parts of the image areas contribute to detect spoofing, we adopted Grad-CAM visualization method (Ramprasaath et al., 2016) to our DCNN models.

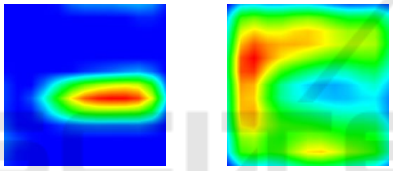
Grad-CAM highlights importance of each

neurons for an each prediction. To obtain the class-discriminative localization map (Fig. 5 (c)(d), Fig. 6 (c)(d)), Grad-CAM calculate the gradient of the score for the class of interest (in this study, “real” or “spoofed”), with respect to CNN feature maps (for example, ‘relu5’ layer of the DCNN(AlexNet)).

Fig. 5 (a) is the original fingerphoto image, and (c) is the Grad-CAM visualization of the image (a). Fig. 5 (b) is the spoofed fingerphoto image, and (d) is the Grad-CAM visualization of the image (b).

From Fig. 5 (a)(c), we can see that our DCNN model detects spot areas of the genuine image, mainly inside of the finger areas. On the other hand, for the spoofed image, the model detects border areas between background areas and finger areas.

(a) genuine (b) spoofed



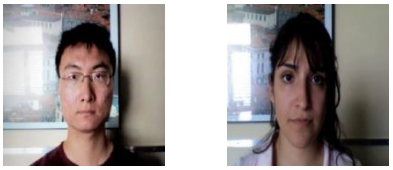
(c) Grad-CAM of (a) (d) Grad-CAM of (b)

Figure 5: (a-b) Original finger image and the generated spoofed image. (c-d) Grad-CAM maps for the original image and the spoofed image.

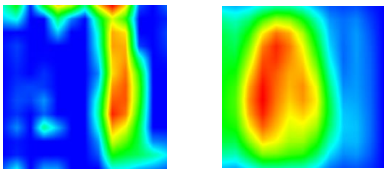
Fig. 6 (a) is the original face image, and (c) is the Grad-CAM visualization of the image (a). Fig. 6 (b) is the spoofed face image, and (d) is the Grad-CAM visualization of the image (b).

From Fig. 6 (a)(c), we can see that our DCNN model detects spot areas of the genuine image, mainly lateral side of the face areas. On the other hand, for the spoofed image, the model detects border areas between background areas and face areas (but mainly background areas), and covers more wider areas, compared to the original (real) image.

Those visualization results suggest that our DCNN models learns the border between background areas and finger/face areas, and utilizes them to detect spoof/genuine images.



(a) genuine (b) spoofed



(c) Grad-CAM of (a) (d) Grad-CAM of (b)

Figure 6: (a-b) Original face image and the generated spoofed image. (c-d) Grad-CAM maps for the original image and the spoofed image.

3.3 Performance Comparison

We compare the performances of the investigated spoofing detection methods (three types of handcrafted features with SVM, and automatic feature extraction/classification by using DCNN (AlexNet)).

For the SVM methods based on handcrafted features, we examined feature extraction + SVM classification times by using linear SVM classifier (liblinear). For the DCNN (AlexNet) based method, we examined input image classification times.

Table 6 shows that DCNN and NRPQA are more than 10 times faster than WLBP or LPQ. The spoofing detection accuracies of those two methods are also much better than the WLBP or LPQ.

Table 6: Summary of the performance (finger).

	Processing time per 1000 images (second)			
	AlexNet	WLBP	NRPQA	LPQ
Feature extraction	4.33	193.0	3.76	49.0
Classification		0.22		
Total	4.33	193.0	3.98	49.22

3.4 Blurriness Tolerant Analysis

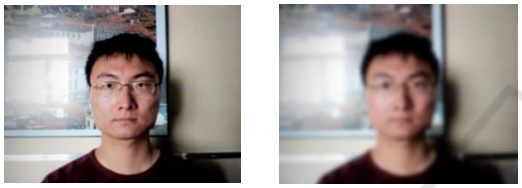
In the task of the spoof detection, presentation images are affected by camera defocus or hand movements, which are commonly seen in both printed photo, replayed video attacks and real faces/fingers.

To simulate image distortions caused by camera defocus or hand movements, we create blurred images by applying image filters (Gaussian blur or motion blur filters) to the test datasets. For the

training images, we do not apply image filters (only use original data sets).

For the Gaussian blurriness, we changed the standard deviation σ parameter, from 0.1 to 2.6 by the step size 0.5. For the motion blurriness, we changed the length of the PSF （point spread function）from 1 to 10 pixels by step size 1, and the filter angle was fixed (11°).

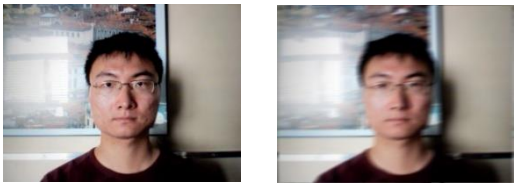
(a) σ =0.1 (b) σ =2.6



(c) σ =0.1 (d) σ =2.6

Figure 7: Examples of Gaussian blurred images (face).

(a) PSF size=1 (b) PSF size=10



(c) PSF size=1 (d) PSF size=10

Figure 8: Examples of motion blurred images (face).

Fig. 13 and Fig. 15 show the spoof detection accuracies for face and spoofed fingerphoto database for Gaussian blurred images. Fig.14 and Fig. 16 show the spoof detection accuracies for face and spoofed fingerphoto database for motion blurred images. In each figure, “NRPQA” represents the method NRPQA defined in the section 2.1 (2), “LBPs” represents the method WLBP defined in the section 2.1 (1), “LPQs” represents the method LPQ defined in the section 2.1 (3) (but combined with the LPQ from wavelet transformed images too), and “DCNN” represents the method defined in the section 2.2. “ALL” represents the features concatenation of all “NRPQA”, “LBPs” and “LPQs”.

(a) σ =0.1 (b) σ =2.6

Figure 9: Examples of Gaussian blurred images (finger).

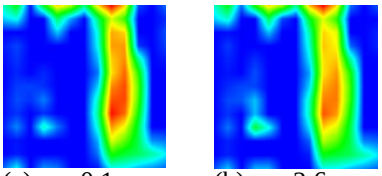
(a) PSF size=1 (b) PSF size=10

Figure 10: Examples of motion blurred images (finger).

In the case of Spoofed Fingerphoto Database, Fig. 13 shows that the more standard deviation of the Gaussian blur increases, the accuracies of the spoof detection decrease steeply. Especially it is prominent for the NRPQA, which exhibited the hight performance for the normal (no additive blurring) images. NRPQA is based on the block noise of the images, which may disappear by the blurring noises. It is also the case for the motion blurring, as you can see in the Fig. 14.

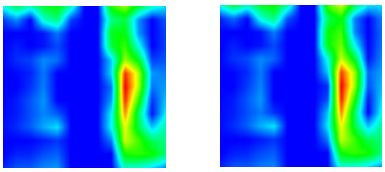
In the case of Replay-Attack Database, Fig. 15 and Fig. 16 also show that the more adding blurring noise, the accuracies of the spoof detection decrease steeply. In this case, the decreases of the accuracies are almost all the same for the features “NRPQA”, “LBPs”, and “LPQs”.

Among the examined methods, DCNN shows not only the highest accuracy but also the highest robustness for the blurring noises. To investigate the stabilities of the examined DCNN model for the increasing intensity of the blurring noise, we adopted Grad-CAM visualization for blurred images. Fig. 11 shows the results of Grad-CAM visualizations for the Gaussian blurred images. Fig. 12 shows the results of Grad-CAM visualizations for the motion blurred images. We can see that highlighted areas are not affected by the increasing intensity of the Gaussian blur, or motion blur. Those results support the results of spoof detection accuracies of the DCNN models.



(a) σ =0.1 (b) σ =2.6

Figure 11: (a) Grad-CAM map for the Gaussian blurred image (σ =0.1). (b) Grad-CAM maps for the motion blurred same image (σ =2.6).



(a) PSF size=1 (b) PSF size=10

Figure 12: (a) Grad-CAM map for the motion blurred image (PSF size=1). (b) Grad-CAM maps for the motion blurred same image (PSF size=10).

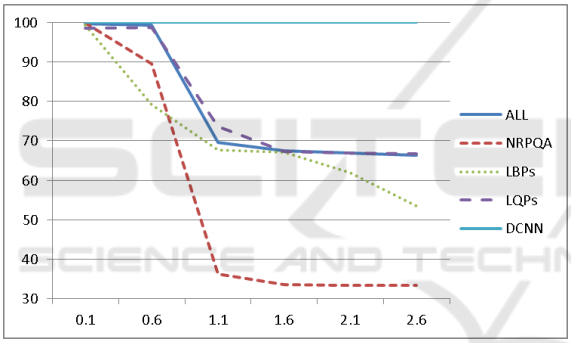


Figure 13: Spoof detection accuracy on the Gaussian blur (finger).

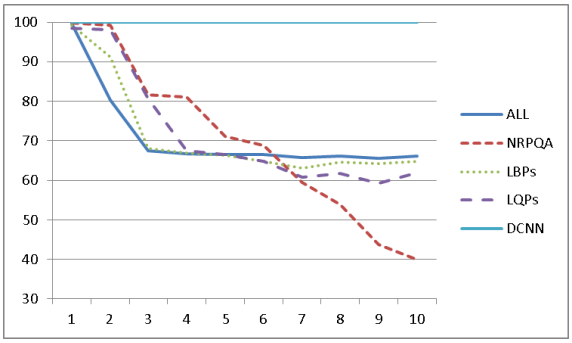


Figure 14: Spoof detection accuracy on the motion blur (finger).

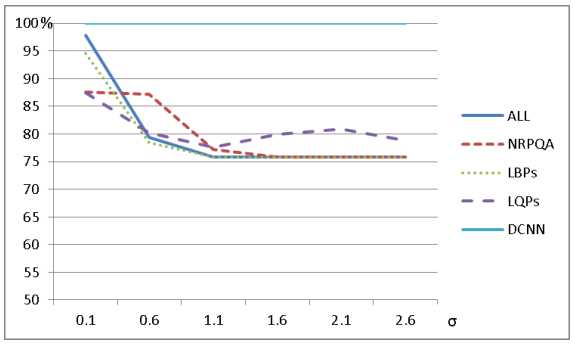


Figure 15: Spoof detection accuracy on the Gaussian blur (face).

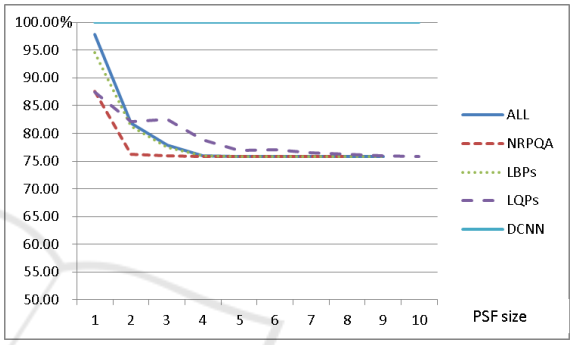


Figure 16: Spoof detection accuracy on the motion blur (face).

4 CONCLUSIONS

In this paper, we address the problem of face/finger spoof detection for the generic camera based biometrics, particularly under noisy conditions. We first propose the anti-spoofing methods based on the local texture features, and achieved less than 1/10 of HTER (half total error rate) compared to the previous methods, for the two different modality databases, Replay-Attack Database (face) and Spoofed Fingerphoto Database (finger).

Furthermore, to simulate the real-life scenarios, we investigate spoof detection under additive noise, such as defocused blurriness and motion blurriness. Our experiments show that using the model trained from clean data, most of the system performance degrades significantly for blurred images. Among the proposed methods, only the DCNN based method shows not only the highest accuracy but also the highest robustness for the blurred images.

For future work on spoof detection of the generic camera based biometrics, we intend to include i) evaluation under the ambient illumination in the use case scenario of interest, ii) investigation for the

cross-modal scenario, iii) develop compact models that can be used on mobile devices.

REFERENCES

Alex, K., Geoffrey, H., 2009. Learning multiple layers of features from tiny images. University of Toronto.

Alex, K., Ilya, Sutskever., Geoffrey.H., 2012. Imagenet classification with deep convolutional neural networks. In *NIPS'12, Advances in Neural Information ProcessingSystems.* NIPS FOUNDATION , Inc.

Archit, T., et al., 2016. Fingerphoto Spoofing in Mobile Devices: A Preliminary Study. In *BTAS'16, International Conference on Biometrics: Theory, Applications, and Systems.* IEEE.

Avinash, K.S et al, Face recognition with liveness detection using eye and mouth movement. In *ICSPCT'14, International Conference on Signal Propagation and Computer Technology*. IEEE.

Bai, J., Ng, T.T., Gao, X., Shi, Y.Q., 2010. Is physics-based liveness detection truly possible with a single image?. In *ISCAS'10*. IEEE.

Di, W., Hu, H., Anil, K.J., 2015. Face spoof detection with image distortion analysis. *IEEE Transactions on Image Process., vol.10, no. 4*. IEEE.

Diogo, C.G., Ricardo. L. de Queiroz., 2015. Face-spoofing 2d-detection based on moire-pattern analysis. *IEEE Transions on Information Forensics and Security, vol. 10, no. 4*. IEEE.

Diego, G., Giovanni P., 2015. An Investigation of Local Descriptors for Biometric Spoofing Detection. *IEEE Transions on Information Forensics and Security, vol. 10, no. 4*. IEEE.

Gang, P., Lin, S., Zhaohui, W., Shihong, L., 2007. Eyeblink-based anti-spoofing in face recognition from a generic web camera. In *ICCV'07, 11th international conference on Computer Vision.* IEEE.

Ivana, C., Andre, A., Sebastien, M., 2012. On the effectiveness of local binary patterns in face anti-spoofing. In *BIOSIG'12, International Conference on Biometrics Special Interest Group*. IEEE.

Javier. G., Sebastien, M., Julian, Fierrez., 2014. Image quality assessment for fake biometric detection: Application to iris, fingerprint and face recognition. *IEEE Transactions on Image Process., vol. 23, no. 2*. IEEE.

Jiangwei, Li., et al., Live face detection based on the analysis of Fourier spectra. In *SPIE'04, Biometric Technology for Human Identification.*

Jianwei, Y., Zhen, L., Stan.Z. L., 2014. Learn convolutional neural network for face anti-spoofing. In CoRR'14, International Conference on Computer Vision and Pattern Recognition. IEEE.

Juho, M., Abdenour, H., Matti, P., 2012. Face spoofing detection from single images using texture and local shape analysis. In IET Biometrics. IET Journals & Magazines.

Keyurkumar, Patel., Hu Han, Anil K. Jain, 2015. *S*ecure Smartphone Unlock: Robust Face Spoof Detection on Mobile. *Technical Report MSU-CSE-15-15*. MSU.

Koichi, I., Takehisa, O., Takafumi, A., 2017. A Study of Anti-Spoofing Using Deep Learning for Face Recognition. In *SCIS '17, 2017 Symposium on Cryptgraphy and Information Security*. IEICE.

Matti, P., Abdenour, H., Guoying, Z., Timo. A., 2011. In *Computer Vision Using Local Binary Patterns.* Springer Science & Business Media.

Xiaofu. H., Yue, L., Pengfei. S., 2009. A new fake iris detection method. In *ICB'09, Advances in Biometrics*. SPRINGER.

Ramprasaath R.S et al, 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. *preprint, arXiv:1610.02391*

Samara, B., Tejas, I.D., Mayank, V., and Richa. S., 2013. Computationally efficient face spoofing detection with motion magnification. In *CVPRW'13, IEEE Conference on Computer Vision and Pattern Recognition Workshop*. IEEE.

Tiago, d.F.P. et al., 2012. LBP-TOP based counter measure against facial spoofing attacks. In *ACCV'12 Workshop on Computer Vision with Local Binary Pattern Variants*. IDIAP.

Tiago, d.F.P. et al., 2014. *Face liveness detection using dynamic texture.* In *EURASIP J. Journal on Image and Video Processing*. SPRINGER.

Timo, A., et al., 2008. Recognition of blurred faces using local phase quantization. In *ICPR'08, 19th International Conference on Pattern Recognition*. IEEE.

Yann, L., Corinna, C., 2010. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/.

Zhiwei, Z., et al., 2012. A face anti-spoofing database with diverse attacks. In *ICB'12, International Conference on Biometrics*. IEEE.

Zhou, W. et al., 2000. Blind Measurement of Blocking Artifacts in Images. In *ICIP2000, Intertational Conference on Image Processing*. IEEE.

Zhou, W., et al., 2002. No-Reference Perceptual Quality Assessment of JPEG Compressed Images. In *ICIP2002, Intertational Conference on Image Processing*. IEEE.