Dual Camera Based Feature For Face Spoofing Detection

Xudong Sun, Lei Huang, and Changping Liu

Institute of Automation
Chinese Academy of Sciences
95 Zhongguancun East Road, 100190, Beijing, China
Email: {sunxudong2013, lei.huang, changping.liu}@ia.ac.cn

Abstract This paper presents a fused feature using dual cameras for face spoofing detection. The feature takes full advantage of input image pairs in terms of texture and depth. It consists of two parts: 2D component and 3D component. For the former, we propose an algorithm based on image similarity to combine every pair of input images into one gray-level image, from which the 2D feature is extracted. For the latter, based on point feature histograms (PFH) method, we describe the point cloud obtained by stereo reconstruction algorithms. The concatenation of 2D and 3D features above is used to represent the input image pair. Experiments on self collected dataset demonstrate the competitive performance and potential of the proposed feature.

Keywords: face spoofing detection, dual cameras, feature fusion, similarity measurement

1 Introduction

Various face recognition systems have been deployed in our daily life, however, traditional face recognition techniques are vulnerable to face spoofing attacks. With digital cameras and cell phones becoming increasingly popular, it is much easier to conduct photo and video spoofing attacks; even 3D face models are used for spoofing. Face spoofing detection, which is aimed to judge the genuine person from fake replicas, plays an important role in security systems and has drawn much attention.

Many anti-spoofing methods have been proposed, which can be roughly classified into four categories: motion based methods [3][9][22][13], texture based methods [2][18][20][14][8], 3D structure based methods [5][6][7][10][19], and fusion methods [22][20].

Motion based methods mainly deal with the human physiological responses or human physical motions, such as eye blinking [22][13], head rotation [3][9], and mouth movement [9]. They can utilize relative features across several frames and they are expected to achieve better results than some other methods, such as texture based features. For some methods, users should strictly follow the instructions. It usually takes a relatively long time to capture image sequences

for spoofing detection, and this kind of methods may be confused or fooled by background motions or replayed video attacks.

Texture based methods analyze skin, reflectance, and other texture properties to classify genuine and fake faces, given that many spoofing faces in photos or videos are different from genuine ones in terms of quality, blurriness, and light condition. Up to now, reflectance information [18][20], imaging banding effects [22], spectral information [14], multi-scale feature [2][23], and many other different kinds of texture based descriptors [7][20][8] are proposed. They can achieve satisfying results, but methods using texture features often do not take adjacent frame information into consideration and may be vulnerable to spoofing attacks made by high quality photos or videos.

3D structure based methods use depth information to distinguish between genuine and spoofing faces. Obviously printed photos or recorded videos show quite different depth structure. Choudhury *et al.* [5] mentioned that depth information can be used to detect planar faces spoofing attacks, but no further experiments were conducted. In [10], based on a binocular framework, the percentage of coplanar facial points was calculated for spoofing detection. It is said to be effective but the framework can not deal with warped spoofing images. Marsico *et al.* [6] presented a system exploiting 3D projective invariant. The system is quite effective and efficient, but the result highly depends on landmark precision. Erdogmus *et al.* [7] gave analyses on various LBP-based anti-spoofing methods using color and depth images obtained from Kinect, which relied on the depth sensors. Wang *et al.* [19] used normal cameras and recovered the sparse 3D shape of faces from image sequences. The recovered depth of landmarks are concatenated to form the final feature, and SVM is used as a classier. The method is able to work on different devices and achieves good performances. However, only depth information of the input images is employed, and other information, such as texture, is ignored.

Nowadays, some researchers are focusing their attentions on fusion methods, trying to make full use of input images. Erdogmus *et al.* [7] used both color and depth images obtained by Kinect in their countermeasures. Yan *et al.* [22] used three scenic clues, including non-rigid motion, face-background consistency and imaging banding effect. In [20], a concatenated feature of four different components (specula reflection, blurriness, chromatic moment, and color diversity) was extracted. Most of the fused clues are among texture and motion features, and 3D structure features are seldom utilized, which may restrain the performance of fusion methods.

With the help of various electronic devices, it is not a hard job to obtain high quality printed photos or deliberately recorded videos, which may even fool the human eyes, so, it may be not an effective way only to use texture descriptor from one image, or only to exploit human responses and motions. In this paper, we use a binocular camera system. Setting up dual cameras and obtaining two images at the same time can provide additional texture and structure information which makes face spoofing detection system more robust and effective. We propose a fused feature which combines both texture and 3D structure clues based on

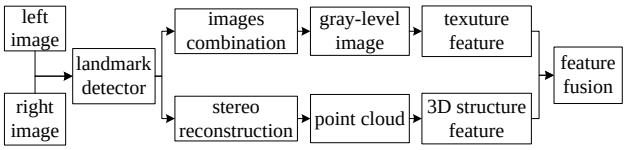


Figure 1. Proposed feature overview

dual cameras. Compared with existing texture based methods, we use binocular images which can provide relative texture clues, and no specific depth sensors are needed in our method. The proposed feature can deal with both warped high quality spoofing images and replayed spoofing videos. Experiments show the promising performance of our proposed feature.

2 System Overview

Motivated by the simple fact that fooling two cameras can not be done simply with captured images or recorded videos any longer, we use dual camera systems to perform spoofing detection, which can capture two images at the same time, to increase the difficulty of face spoofing. Dual cameras are used in proposed framework. We use two cameras of the same type and mount them side by side. The two cameras are in the same direction and they are several centimeters apart. Stereo camera calibration [24] is performed before data collection process.

Our framework is illustrated in Fig. 1. The proposed feature consists of two main parts. At first, we combine left and right images and obtain one 2D gray-level image using similarity measures, and Gabor feature [11] is extracted from regions of interest in the similarity image. Then, using reconstructed sparse 3D facial structure by stereo reconstruction algorithms [24], a simplified PFH method based on PFH [17] and FPFH [16] is used to extract point cloud features. Finally, 2D texture features and 3D depth features are both normalized before concatenated into one vector feature, and we use this fused feature to represent the input image pair.

3 Proposed Feature

As we all know, the depth of genuine faces is quite sophisticated: eyes are sunken; noses are close to cameras, and ears are relatively far. So, the distances do not change smoothly especially around organs such as eyes and noses. The irregular surface contributes to complicated occlusion in face images and face images vary as view direction changes. In contrast, the depth is almost flat for printed or replayed spoofing faces; it changes smoothly even if the photo is deliberately warped. The gradually changed surface only leads to simple occlusion.

With the help of binocular camera images, we can easily analyze detailed texture differences caused by occlusion in face areas. The genuine faces are likely

to have different appearances, especially where surface alters much, while spoofing faces may not contain such characteristics. We utilize this clue to construct proposed 2D-texture feature.

It should be noted that genuine and spoofing faces have quite various depth information. To make proposed methods robust and make full use of input image pairs, we also employ 3D-depth features based on depth values of the landmarks. We have binocular framework, so we can obtain more than merely texture feature or depth feature. As discussed earlier, every single feature has its own drawbacks and the fusion feature may work better, so, it may be a wise idea to utilize both features to perform spoofing detection. As shown in our experiment in Sec 4, the fusion method achieves better results indeed. Hence, in our system, both 2D and 3D features are exploited together to conduct final classification.

3.1 2D-Texture Feature

To extract the detailed occlusion differences in binocular images, we first adopt a similarity descriptor and combine binocular images into one gray-level image; then, we choose Gabor wavelet filter [11] to extract the differences.

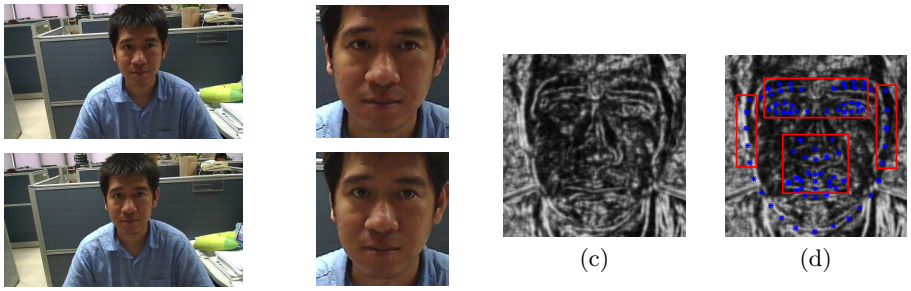
Our similarity descriptor approach is inspired by BRIEF [4] and ORB [15] feature which both show that image patches can be used to extract effective and efficient local features. BRIEF [4] is an efficient feature point descriptor, which uses binary strings features and Hamming distance, and ORB is one of popular descriptors based on BRIEF. However, both descriptors are initially designed for key points only, and for two adjacent pixels, their descriptors may be similar and can not reflect possible diversities. So we propose a modified BRIEF feature descriptor M-BRIEF with Hamming distance to describe similarities for every corresponding pixel, which will be shown later.

First of all, we preprocess input images. A pre-trained supervised descent method [21] is used to locate landmarks on input faces. Then, input images are registered using eye coordinates. The normalized face images are cropped to the same size. To make the descriptor noise-insensitive, we use a Gaussian filter to smooth the normalized images.

Next, the modified BRIEF descriptors of all pixels are supposed to be calculated. For every point $\mathbf{p} = (u, v)^T$ in left normalized image, we create a patch $\mathbf{P}$ with the size $S \times S$ around it, and $\mathbf{x}$ is one of the points in the patch. At the same time, we can find a corresponding point $\mathbf{p}'$ with the same coordinate $(u, v)^T$ in right normalized image. A patch $\mathbf{P}$ can also be created and $\mathbf{y}$ is a point in the patch. Now, just like BRIEF descriptor [4], we define a similar test function τ on patch $\mathbf{P}$ as:

$$\tau(\mathbf{P}; \mathbf{x}, \mathbf{y}) := \begin{cases} 1 & \text{if } \mathrm{sum}(\mathbf{x}) < \mathrm{sum}(\mathbf{y}) \\ 0 & \text{otherwise} \end{cases} . \tag{1}$$

Where $\mathrm{sum}(\mathbf{x})$ is the sum of intensities of pixel $\mathbf{x}$ and its eight surrounding pixels. Here we calculate the sum of surrounding pixels instead of only one pixel $\mathbf{x}$ in [4] and [15] to make it robust for potential errors in locating landmarks and



(a) (b) (c) (d)

Figure 2. Image combination process. (a) shows input left and right images, (b) shows normalized face images, (c) is combined similarity image, (d) illustrates the chosen regions

registering faces. Now we randomly choose one set of n_d $(\mathbf{x}, \mathbf{y})$-location pairs, where $\mathbf{x}$ is in the left image patch and $\mathbf{y}$ is in the left image patch. We use the same set of n_d location pairs for patches of all pixeles.

In this way we uniquely define a set of binary tests. Thus the modified BRIEF descriptor of point $\mathbf{p} = (u, v)^T$ becomes one n_d-dimensional bit string:

$$\text{M-BRIEF}(\mathbf{p}) := \sum_{1 \le i \le n_d} 2^{i-1} \tau(\mathbf{P}; \mathbf{x}_i, \mathbf{y}_i)\,. \tag{2}$$

We use the same parameters as mentioned in [4], where Gaussian kernel is set to 2, patch size $S \times S$ is set to 9×9 and n_d is set to 256.

After calculating M-BRIEF descriptor of all pixels, we use Hamming distance as similarity metrics between every pair of corresponding pixels with the same coordinate (u, v) in binocular images. Then, the distance is normalized to $[0, 1]$, denoted as $\mathrm{D}(u, v) \in [0, 1]$. Right now, we can simply get gray-level similarity image by $\mathrm{S}(u, v) = \mathrm{floor}(\mathrm{D}(u, v) \times 255)$. The whole process of getting combined similarity image is illustrated in Fig. 2.

Pixels in combined gray-level images represent similarity distances between corresponding pixels of binocular images. The whiter the pixels are, the larger the differences exist in binocular images. As analyzed before, texture differences of binocular images exist especially where the depth changes dramatically, and the white pixels represent potential texture differences. For genuine faces, there are certainly more changes in depth, so, these bright white pixels may probably represent potential genuine faces.

Fig. 3 shows more examples of constructed similarity images. It can be inferred that, in face areas from top row images, similarity images of spoofing ones contain relatively less bright white pixels, and streaks and organ outlines are also simpler and thinner, especially those around noses and eyes. This satisfies our hypotheses, but it is still quite hard for humans to make the correct judgements directly from similarity images. So, we adopt further feature extraction and construct our 2D-texture feature.

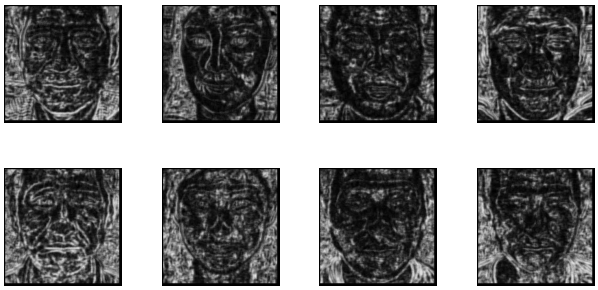


Figure 3. Examples of constructed similarity images of spoofing faces (top row) and genuine faces (bottom row). However, in reality it is quite hard to tell them apart by humans directly.

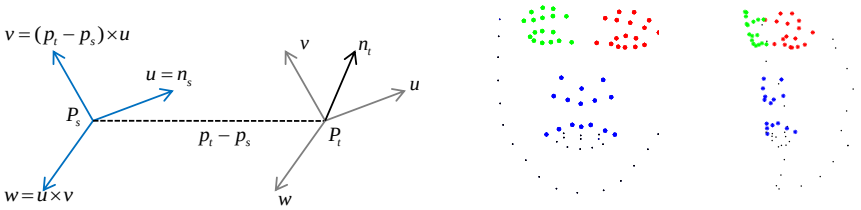
From combined similarity images, we manually choose several regions of interest, which are considered to contain some major differences between genuine face images and spoofing ones. In this paper, based on our analyses and experiments, we choose 4 simple regions according to face landmarks, which roughly include regions around noses, mouths, eyes, and ears. We mainly choose those regions in which depth of genuine persons may alter suddenly, and regions where depths varies gradually, such as cheeks, are not used. The chosen regions are illustrated in red rectangles in Fig. 2(d).

Finally, we use Gabor wavelet filter [11] in eight directions and five scales, to extract feature vectors in each region. Downsampling process is also adopted in our system and downsampling factor is set to 64 as in [11] to perform feature reduction. Downsampled feature vectors of all chosen regions are concatenated together forming a feature vector, but Gabor texture feature is stll too long even after downsampling process. So, we employ traditional Principal Component Analysis (PCA) for dimension reduction, and only main energy is preserved, forming our final 2D-texture feature. For images in testing process, we use the same PCA parameters as those in training process.

3.2 3D-Depth Feature

Before obtaining proposed depth feature, we utilize calibrated cameras and stereo construction algorithms [24] to get depth information. In our proposed method, only depths and normal vectors of some landmarks are used, forming a sparse point cloud. The following depth features are extracted from the sparse ponit cloud we obtain.

Point feature histograms (PFH) [17] is a powerful 3D point feature, and it is used in many areas such as point cloud matching. We are inspired by PFH and FPFH [16] algorithms, and modify and simplify PFH algorithm into extracting features of sparse point cloud in our case. The main modifications of PFH [17] and our main algorithmic processes are listed as follows:



(a) Darboux *uvn* frame (b) front view (c) profile view

Figure 4. (a) is illustration of Darboux *uvn* frame, (b) and (c) are front and profile views of 3D landmark structure used in proposed method. Different colors means different sets, and small points in black are not used.

1. Note that we have a sparse point cloud, rather than using k-neighborhood of a specific point [17], we group landmarks manually into several sets, as illustrated in Fig. 4 (b) (c). To make our methods less noise sensitive, we omit the points on the face contour which may not be so precise, and mainly construct three sets around eyes and nose. Our 3D feature is extracted from each point set respectively.

   For point pairs $< p_s, p_t >$ in each set, their estimated surface normal vectors n_s, n_t and the Darboux *uvn* frame are defined exactly the same way as in [17] ($u = n_s, v = (p_t - p_s) \times u, w = u \times v$), which are illustrated in Fig. 4 (a). It should be noted that given two sequential points $< p_s, p_t >$ and their corresponding normal vectors, the *uvn* frame will be uniquely defined.
2. The scale of faces does not change so much under our circumstance, we just omit the Euclidean distance component in original PFH method. In every set of points, we extract three primary features from point pairs, and we can get bin histogram as our 3D-feature, where each bin at index idx contains the *percentage* of the point pairs defined by idx:

$$\left.\begin{array}{r} f_1 = v \cdot n_t \\ f_2 = w \cdot n_t \\ f_3 = \frac{(u \cdot (p_t - p_s))}{|p_t - p_s|} \end{array}\right\} idx = \sum_{i=1}^{i \leq 3} \text{step}(thre_i, f_i)\,. \tag{3}$$

   where $\text{step}(thre_i, f_i)$ is defined as 1 if $thre_i > f_i$ and 0 otherwise.

By setting an appropriate threshold $thre_i$, every primary feature can be turned into two categories and in this way we can obtain $2^3 = 8$ histogram bins for each depth feature. In every set of our sparse points, we calculate the histogram of all point pairs, and we can get an eight-dimension histogram feature. The final 3D-feature can be gained by concatenating histogram bins in all manually grouped sets. We have 3 sets of points in total as illustrated in Fig. 4 (b), so the final dimension of proposed 3D-feature is $3 \times 8 = 24$.

Figure 5. Some illustrations of our dataset. Column 1 and column 3 are from left view, and column 2 and column 4 are from right view respectively.

4 Experiments

4.1 Dataset Description

As far as we know, there are no public binocular datasets for spoofing detection. So, we use dual cameras to collect data ourselves. The two cameras are both calibrated respectively, and stereo calibration [24] is performed. There are 35 users taking part and we allow them to move or rotate heads as they like. For each user, we use his or her high quality color photos to do the same thing imitating face spoofing attacks, and printed photos can be rotated or warped freely. We also record videos for all users and use these videos to perform tablet replay attacks. For those wearing glasses, the same goes for their printed photos and recorded videos. In our dataset, one image pair consists of one image from left camera and one image from right camera. The numbers of image pairs are shown in Table 1, and some examples are shown in Fig. 5.

Table 1. The number of image pairs in our dataset.

	Genuine	Photo Spoofing	Tablet Spoofing
Number of pairs	4869	4758	5674

4.2 Baseline Method

The original BRIEF and PFH descriptor are not designed for the situations like ours as mentioned before, so, we have to choose our baseline methods. There

are only a few methods dealing with dual cameras for face spoofing detection, and spoofing photos may be warped in our dataset. We choose depth method mentioned in [19] as one of baseline methods. They proposed an algorithm that recovered the 3D sparse facial structure, and they used all 3D coordinates as a feature vector. We just calculate depth by stereo reconstruction algorithms and use the same number of landmarks as that in [19].

According to recent research [8], although many LBP based methods are proposed relatively a long time before, they are still powerful local texture descriptors in terms of face spoofing detection, such as CoA-LBP [12]. Many of the methods have public access code [8], which makes it easier and more convincing to make comparisons. Besides CoA-LBP, multiscale local binary pattern (MsLBP) and its related methods draw much attention in previous works such as [2][23]. In this paper, we find MsLBP can achieve a bit better results on our collected dataset than CoA-LBP with a linear SVM classifier, so we utilize MsLBP from [23] on every single image from our dataset as another baseline method.

4.3 Results and Discussions

As used in [8], we also employ a simple linear Support Vector Machine (SVM) as final classifier, and leave-one-out configuration is adopted. More specifically, we use images of 34 persons to train SVM model; then, we test the model on images of remaining person.

We adopt traditional False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER), and Half Total Error Rate (HTER) to evaluate these methods. Besides, based on the metric developed in ISO/IEC 30107-3 [1], we also report Attack Presentation Classification Error Rate (APCER), Normal Presentation Classification Error Rate (NPCER), and Average Classification Error Rate (ACER). In our experiment tables, FAR, FRR, APCER, and NPCER are the exact values when the highest SVM classification accuracy is reached, and all the evaluation metrics are mean values according to leave-one-out configuration.

Table 2. Results on photo spoofing images(%)

Methods	FAR	FRR	HTER	APCER	NPCER	ACER	EER
Proposed Method	0.99	2.00	1.50	1.13	1.76	1.44	0.76
Proposed Texture	6.10	4.19	5.15	7.21	3.53	5.37	3.52
Proposed Depth	1.83	2.85	2.34	2.09	2.50	2.29	1.63
MsLBP [23]	14.78	8.91	11.85	18.56	6.93	12.74	6.71
3D Coordinates [19]	1.75	5.59	3.67	1.95	5.05	3.50	2.33

The results on both photo and tablet spoofing images between proposed method and baseline methods are shown in Table. 2 and Table. 3 especially. We

not only evaluate the results of our proposed methods, but also examine our proposed 2D and 3D feature alone.

From both tables, we can see multiscale local binary pattern does not perform as desired. The main reason is that we use high quality images and videos; using texture alone methods such as MsLBP may not be a good idea. Besides, MsLBP can only perform on one single image, and binocular information in our dataset is not utilized at all, which is one of the disadvantages of traditional texture based spoofing detection methods.

The 3D depth based methods, which utilize binocular images to recover the depth information, perform much better than MsLBP. However, we do not get as excellent results as that in [19], in which all the images are classified correctly. It may be explained that by following reasons: we reconstruct depths only based on one left and one right image, so stereo correspondence algorithm may easily be affected by some factors, such as the glasses, bangs hair, and pale faces. These factors all leads to noisy or even mistaken depth values, which worsen the results directly. We have checked the reconstructed results of our dataset manually, and we have found for a minority of persons, especially for some female persons wearing glasses and bangs hair, the recovered depth is not so ideal. For examples, we find that results of depth based methods on persons among the first two rows in Fig. 5 are much worse than average values, which are not listed here though. So using depth based feature alone may not produce very good results and it might be a wise choice to use the fused feature instead.

Table 3. Results on tablet spoofing images(%)

Methods	FAR	FRR	HTER	APCER	NPCER	ACER	EER
Proposed Method	0.78	1.21	0.99	0.66	1.42	1.04	0.63
Proposed Texture	4.65	3.32	3.99	5.45	2.82	4.14	3.34
Proposed Depth	1.21	1.47	1.34	1.03	1.73	1.38	0.73
MsLBP [23]	7.72	8.20	7.96	6.48	9.74	8.11	6.06
3D Coordinates [19]	2.30	2.38	2.34	2.66	2.06	2.36	1.49

Comparing results between Table. 2 and Table. 3, it can be easily concluded that almost all methods perform better on photo spoofing images than tablet spoofing ones. The spoofing media contributes a lot to such results. Images displayed by tablet screen is always flat and screens can not be warped at all, which benefits depth based methods. In addition, the screen emits lights itself so there are no interferences by environment lighting such as shadows, which makes texture methods relatively easier.

We also evaluate methods above on all collected images, imitating the situation in which we have no idea about how the spoofing detection systems will be challenged. The experiment results are shown in Table. 4, and DET curves are illustrated in Fig. 6. It should be noted that although both proposed 2D-texture

Table 4. Results on all collected images(%)

Methods	FAR	FRR	HTER	APCER	NPCER	ACER	EER
Proposed Method	1.06	1.11	1.09	0.51	2.28	1.40	0.58
Proposed Texture	5.62	3.90	4.76	6.62	3.30	4.96	3.54
Proposed Depth	2.04	2.62	2.33	0.97	5.42	3.20	1.83
MsLBP [23]	18.63	9.49	14.06	9.07	19.41	14.24	10.52
3D Coordinates [19]	2.21	1.93	2.07	2.06	2.37	2.22	1.75

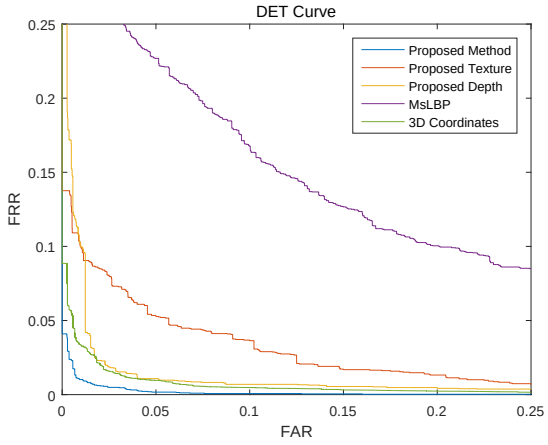


Figure 6. Results obtained on all collected images.

and 3D-depth methods do not have overwhelming advantages over depth based baseline method [19], the fusion method obtains quite satisfied results with EER of 0.68% and ACER of 1.40%.

5 Conclusion

In this paper, we propose a novel feature for face spoofing detection using dual cameras. On one hand, we investigate image texture details caused by different surface situations between genuine and spoofing faces. We combine a pair of binocular images into one gray-level image, on which texture features are extracted. On the other hand, we extract depth feature by using a sparse point cloud with limited number of points. Experiment results demonstrate the promising performance of the fused feature. In the future, we will focus on dealing with different kinds of spoofing attacks, such using 3D masks or 3D head models.

References

1. ISO/IEC 30107-3 Biometric presentation attack detection - Part 3: Testing and reporting. International Organization for Standardization (2015)
2. Arashloo, S.R., Kittler, J., Christmas, W.: Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features. IEEE Transactions on Information Forensics and Security 10, 2396–2407 (2015)
3. Bao, W., Li, H., Li, N., Jiang, W.: A liveness detection method for face recognition based on optical flow field. In: International Conference on Image Analysis and Signal Processing. pp. 233–236. IEEE (2009)
4. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: European Conference on Computer Vision. vol. 6314, pp. 778–792. Springer (2010)
5. Choudhury, T., Clarkson, B., Jebara, T., Pentland, A.: Multimodal person recognition using unconstrained audio and video. In: International Conference on Audio- and Video-Based Person Authentication. pp. 176–181. Citeseer (1999)
6. De Marsico, M., Nappi, M., Riccio, D., Dugelay, J.L.: Moving face spoofing detection via 3d projective invariants. In: IAPR International Conference on Biometrics. pp. 73–78. IEEE (2012)
7. Erdogmus, N., Marcel, S.: Spoofing in 2d face recognition with 3d masks and anti-spoofing with kinect. In: International Conference on Biometrics: Theory, Applications and Systems. pp. 1–6. IEEE (2013)
8. Gragnaniello, D., Poggi, G., Sansone, C., Verdoliva, L.: An investigation of local descriptors for biometric spoofing detection. IEEE Transactions on Information Forensics and Security 10(4), 849–863 (2015)
9. Kollreider, K., Fronthaler, H., Bigun, J.: Evaluating liveness by face images and the structure tensor. In: Workshop on Automatic Identification Advanced Technologies. pp. 75–80. IEEE (2005)
10. Li, Q., Xia, Z., Xing, G.: A binocular framework for face liveness verification under unconstrained localization. In: International Conference on Machine Learning and Applications. pp. 204–207. IEEE (2010)
11. Liu, C., Wechsler, H.: Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. IEEE Transactions on Image Processing 11, 467–476 (2002)
12. Nosaka, R., Ohkawa, Y., Fukui, K.: Feature extraction based on co-occurrence of adjacent local binary patterns. In: Pacific-Rim Symposium on Image and Video Technology. pp. 82–91. Springer (2011)
13. Pan, G., Sun, L., Wu, Z., Wang, Y.: Monocular camera-based face liveness detection by combining eyeblink and scene context. Telecommunication Systems 47, 215–225 (2011)
14. Pinto, A., Pedrini, H., Robson Schwartz, W., Rocha, A.: Face spoofing detection through visual codebooks of spectral temporal cubes. IEEE Transactions on Image Processing 24(12), 4726–4740 (2015)
15. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: International Conference on Computer Vision. pp. 2564–2571. IEEE (2011)
16. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: International Conference on Robotics and Automation. pp. 3212–3217. IEEE (2009)

17. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Aligning point cloud views using persistent feature histograms. In: International Conference on Intelligent Robots and Systems. pp. 3384–3391. IEEE (2008)
18. Tan, X., Li, Y., Liu, J., Jiang, L.: Face liveness detection from a single image with sparse low rank bilinear discriminative model. In: European Conference on Computer Vision. vol. 6316, pp. 504–517. Springer (2010)
19. Wang, T., Yang, J., Lei, Z., Liao, S., Li, S.Z.: Face liveness detection using 3d structure recovered from a single camera. In: International Conference on Biometrics. pp. 1–6. IEEE (2013)
20. Wen, D., Han, H., Jain, A.K.: Face spoof detection with image distortion analysis. IEEE Transactions on Information Forensics and Security 10, 746–761 (2015)
21. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: Computer Vision and Pattern Recognition. pp. 532–539. IEEE (2013)
22. Yan, J., Zhang, Z., Lei, Z., Yi, D., Li, S.Z.: Face liveness detection by exploring multiple scenic clues. In: International Conference on Control Automation Robotics & Vision. pp. 188–193. IEEE (2012)
23. Yang, J., Lei, Z., Liao, S., Li, S.Z.: Face liveness detection with component dependent descriptor. In: Biometrics (ICB), 2013 International Conference on. pp. 1–6. IEEE (2013)
24. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 1330–1334 (2000)