

Research Article

Face Spoof Attack Detection with Hypergraph Capsule Convolutional Neural Networks

Yuxin Liang, Chaoqun Hong*, Weiwei Zhuang

School of Computer Science and Information Engineering, Xiamen University of Technology, #600 Ligong Road, Jimei Xiamen, 361024, China

ARTICLE INFO*Article History*Received 26 May 2020
Accepted 12 Apr 2021*Keywords*Face spoof attack detection
Multiple-feature learning
Capsule neural networks
Hypergraph regularization**ABSTRACT**

Face authentication has been widely used in personal identification. However, face authentication systems can be attacked by fake images. Existing methods try to detect such attacks with different features. Among them, using color images become popular since it is flexible and generic. In this paper, a novel feature representation for face spoof attack detection, namely hypergraph capsule convolutional neural networks (HGC-CNNs), is proposed, which takes advantage of multiple features. To achieve it, capsule neural networks are used to integrate different types of features. In addition, hypergraph regularization is applied to learn the correlations among samples. In this way, the descriptive power is improved. The proposed feature representation is compared with existing features for face spoof attack detection and experimental results on two commonly used datasets emphasize the effectiveness of HGC-CNN.

© 2021 The Authors. Published by Atlantis Press B.V.

This is an open access article distributed under the CC BY-NC 4.0 license (<http://creativecommons.org/licenses/by-nc/4.0/>).**1. INTRODUCTION**

In the past few years, biometric identification technologies have been widely applied for authentication. Among them, face recognition attracts plenty of attention since it is contactless and easy to use. Therefore, it is used in lots of applications such as entrance guard, mobile payment and so on. Since authentication with face recognition becomes popular, security is critical in these applications. Faces are important privacies of people. However, it is easy to capture facial images or make fake images. If they are abused by malicious attackers, it is possible to conduct face spoof attacks to face recognition systems. In this way, the applications with facial authentication are vulnerable.

To avoid face spoof attacks, researchers make various attempts, which are called face spoof attack detection or face liveness detection. The key is how to distinguish real faces and fake photos (videos). A straightforward method is using motion cues. Some applications such as mobile payment and online banking systems may require the users to do some specific actions. These actions can be recognized with existing landmark detection methods [1]. In this way, the success rate of spoof attacks are significantly reduced. However, it takes much time to pass the authentication. Besides, in many applications, the users may not cooperate with the requirements. Therefore, some methods without interactions are needed.

One choice of face spoof attack detection without interactions is still using motion features. These motion-based methods are used with videos. Anjos *et al.* used optical flow to explore the foreground/

background motion correlation [2] and it outperformed previous methods using motion analysis. Bharadwaj *et al.* used motion magnification to enhance the facial expressions in videos [3]. Then, facial images are represented by multi-modal features. They are local binary pattern (LBP) in images and histogram of oriented optical flow (HOOF) in videos.

On the other hand, another choice is used with static images. Some of them requires additional sensors. For example, Liu and Kumar used both color images and near infrared illumination to detect face spoof attacks with using 3D face masks [4]. Convolutional neural networks (CNNs)-based configurations were investigated to improve the performance. Wang *et al.* proposed to use the depth information extracted by a Kinect sensor [5]. Depth information was integrated with 2D textual information learned from 2D facial image regions. Similar to the aforementioned method, textual information is also extracted with CNN. Bresan *et al.* proposed to combine intrinsic image properties and deep neural networks [6]. Depth, saliency and illumination maps are used as features and a meta-learning method is used as the classifier. Since additional sensors are not always available, face spoof attack detection with color images are much more commonly used. More of the researches focus on image features describing different characteristics between real faces and fake photos. Based on improvements of LBP, Peng *et al.* proposed guided scale-based local binary pattern (GS-LBP) and local guided binary pattern (LGBP) [7]. In this way, textual features are extracted and support vector machine (SVM) is used for classification. Galbally and Marcel transmitted the problem of spoof attack to image quality assessment [8]. They used four different features for image distortion analysis.

*Corresponding author. Email: cqhong@xmut.edu.cn

Due to the ability on concept abstraction and description, deep learning is also used in face spoof attack detection. Yang *et al.* firstly proposed to use neural networks for spoof feature detection and SVM for classification [9]. Feng *et al.* combined Shearlet features, RGB images and optical flow in neural networks [10]. In this way, three different features for image quality, pixel colors and motion cues are used. Lei *et al.* divided faces into parts and used the inherent representation in CNN as features [11]. Atoum *et al.* used HSV and YCC images instead of RGB images. A two-steamed strategy is used to detect features in color images and depth images. Liu *et al.* used both spatial and temporal features in CNN for attack detection [12]. To detect unseen attacks, Shao *et al.* focused on the domain generalization problem [13]. They made use of meta-learning and the feature space is regularized by the trained domain knowledge.

According to the review above, we can figure out the trend of face spoof attack detection is using color images and applying multiple features in order to improve universality [14–16]. However, existing methods usually use traditional image features and ignore the semantic information. In this paper, we propose a method for face spoof attack detection with hypergraph capsule convolutional neural networks (HGC-CNNs). It improves descriptive power with correlations of multiple samples and integration of multiple features. The contribution can be summarized as:

1. First, a novel method for feature representation is proposed for face spoof attack detection. It is based on the combination of graph convolutional neural networks (GCNNs) and capsule networks. Correlations of multiple samples are learned with GCNNs, while multiple features are integrated in capsules.
2. Second, the process of graph construction is further improved with hypergraph regularization, in which different types of features are represented in a uniform feature space. Adjacency can be computed and applied in construction of graph Laplacian.
3. Finally, based on the novel representations, face spoof attack is detected with classification. Comprehensive experiments

are conducted to validate our method on two commonly-used benchmark datasets. The experimental results indicate the improvement of the proposed method.

The remainder of our paper is organized below. In Section 2, we outline the proposed HGC-CNN first and then introduce it in detail. After theoretical details, in Section 3, we show the performance of HGC-CNN on face spoof attack detection by experimental evaluations and comparisons. Finally, in Section 4, we come to the conclusion.

2. HYPERGRAPH CAPSULE CONVOLUTIONAL NEURAL NETWORKS

2.1. Outline

The overall process of the proposed method is shown in Figure 1. First, different features of facial images are extracted. In our implementation, we use color pixels, LBP and image quality [8]. The framework is flexible and features can be easily added or removed. Second, a capsule layer is used, which is the key contribution of the proposed method. In this layer, different features are enclosed to capsules and then they are integrated with hypergraph regularization. Third, the integrated features are further processed by three full connected layers to obtain the final feature representation, which is similar to the implementation of the original capsule networks.

2.2. Graph Regularized Capsule Convolutional Neural Networks

CNNs have proven successful in image description due to their high-level semantic abstraction capabilities. Let $X = \{x_i | i = 1, 2, \dots, N\}$ be a set of input samples. In traditional CNNs with \mathcal{L} layers, the for-

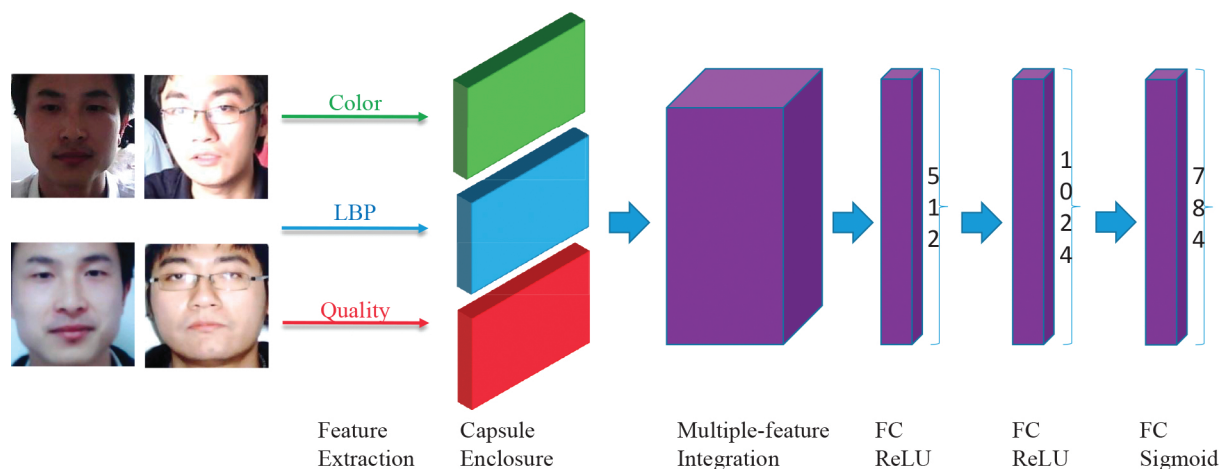


Figure 1 | The flowchart of face spoof attack detection with hypergraph capsule convolutional neural networks. (1) Face detection is performed to obtain facial images. (2) The features of facial images are extracted. Different types of features can be used, such as color features, LBP and image quality. (3) Different features are enclosed in capsules. (4) Different types of features are integrated with hypergraph regularization. (5) Three FC layers are used to compute the hidden representations. They can be used in classification with different classifiers, such as SVM, softmax and so on.

ward output on x_i on the l -layer can be defined as

$$(x_i)^l = f\left((W)^l(x_i)^{l-1}\right), \quad (1)$$

where $f(\cdot)$ is widely-known as the activation function and $(W)^l$ is the mapping parameters of the l -th layer.

There are several drawbacks on the traditional definition of CNNs. The first one is CNNs ignore the semantic correlations among samples. Therefore, Kipf and Welling proposed GCNNs [17]. In GCNNs, the output on x_i depends on its K neighbors, which is given by

$$(x_i)^l = \sum_{k=1}^K a_{ik} f\left((W)^l(x_j)^{l-1}\right), \quad (2)$$

where a_{ik} denotes the adjacency weights between nodes i and k . It is usually normalized.

The second drawback of traditional CNNs is they simply work on scale values, which limits the descriptive ability. To improve it, Sabour *et al.* proposed capsule neural networks [18]. In capsule networks, the input is formed as a vector. This vector is called a capsule and more properties can be contained in it. In this way, highly informative outputs can be computed with

$$(x_i)^l = \begin{bmatrix} f\left((W)^l(x_i^1)^{l-1}\right) \\ f\left((W)^l(x_i^2)^{l-1}\right) \\ \dots \\ f\left((W)^l(x_i^P)^{l-1}\right) \end{bmatrix}, \quad (3)$$

where P is the types of properties in one capsule.

Inspired by GCNN and capsule networks, the proposed graph-regularized capsule convolutional neural networks (GRC-CNNs) combines the above two ideas and propose a novel approach for multiple-feature learning. The key forward operation is defined as

$$(x_i)^l = \begin{bmatrix} \sum_{j=1}^k a_{ij} f\left((W)^l(x_j^1)^{l-1}\right) \\ \sum_{j=1}^k a_{ij} f\left((W)^l(x_j^2)^{l-1}\right) \\ \dots \\ \sum_{j=1}^k a_{ij} f\left((W)^l(x_j^P)^{l-1}\right) \end{bmatrix}. \quad (4)$$

Then, the general form of GRC-CNN can be rewritten as

$$(X)^l = L(W)^{l-1}(X)^{l-1}, \quad (5)$$

L is known as the graph Laplacian matrix and contains the semantic correlations among samples in X .

2.3. Hypergraph Construction

In the scenario of using multiple features, simple summing is used in the traditional routine of capsule networks. According to it, Eq. (5) can be transformed to

$$(X)^l = \sum_{p=1}^P L_p(W_p)^{l-1}(X_p)^{l-1}, \quad (6)$$

However, in the proposed method, we propose multiple-feature integration by hypergraph regularization. In traditional graph regularization methods, the relationships among features are assumed to be pairwise, which is over-simplified and limits the descriptive power. In hypergraph, features with the same property are connected by hyperedges. With hyperedges, more than two features can be connected. Therefore, the key to obtain good performance is the hypergraph construction of a uniform L which integrates multiple L_p .

To simplify the notation, \overline{X}_p is used to represent the output of X_p instead of $W_p X_p$, which can be assumed to be new features. To integrate them, we use patch alignment framework [19,20]. In this framework, feature vectors can be represented by vertices in this graph. Notations are shown in Table 1 and the process of multiple-feature integration is shown in Figure 2. In details, the Laplacian matrix L_p for p -th features and the global L can be computed with two stages.

1. Part optimization stage: We define one patch to be the vertices connected by one hyperedge. Thus, a patch in this graph is defined by

$$\arg \min_{f \in \mathbb{R}^{|V|}} \sum_{m,n \in e} \frac{w(e)}{\delta(e)} \left(\frac{y_m}{\sqrt{d_m}} - \frac{y_n}{\sqrt{d_n}} \right)^2 \quad (7)$$

Table 1 Notations in the hypergraph.

Symbol	Definition
u, v	Vertices in the hypergraph
e	Edges in the hypergraph
$\omega(e)$	The weight of an edge e
$\delta(e)$	The degree of an edge, e . It illustrates how many vertices are connected by e . In traditional graph representation, $\delta(e) = 2$.
$d(v)$	The degree of a vertex, v . It is calculated by summing the weighting values of edges connected to this vertex.
DV	The diagonal matrix containing the vertex degrees
DE	The diagonal matrix containing the edge degrees
H	In this matrix, $H(v, e) = 1$ if $v \in e$
Ω	The diagonal matrix containing the weights of hyperedges
V	The set of vertices
E	The set of edges
Y	The label of vertices

To solve the above minimal problem, for one patch, we need to compute

$$\sum_{m,n \in e} \frac{w(e)}{\delta(e)} \left(\frac{y_m}{\sqrt{d_m}} - \frac{y_n}{\sqrt{d_n}} \right)^2, \quad (8)$$

In Eq. (8), we use all pairs of vertices contained by a hyper-edge, e , and compute the summing value of

$$\frac{w(e)}{\delta(e)} \left(\frac{y_m}{\sqrt{d_m}} - \frac{y_n}{\sqrt{d_n}} \right)^2. \quad (9)$$

By expanding Eq. (9) and combining items, the patch optimization formulation for each hyperedge can be defined as

$$L_p = \frac{1}{2} \sum_{v \in e} \frac{Y}{DV_v^2} H'_e \frac{\Omega}{DE} H_e \frac{Y}{DV_v^2}, \quad (10)$$

where $Y = y_i \mid 1, 2, \dots, n$ is the labels of images. For out-of-sample images, Y can be estimated by KNN. Besides, Matrix E is defined as

$$\begin{bmatrix} -\vec{e}^T \\ I \end{bmatrix} \quad (11)$$

where $\vec{e} = [1, \dots, 1]^T$, I is an $n \times n$ identity matrix. The computational complexity of this stage is $K \times n \times d^2$.

- Whole alignment stage: In the hypergraph, the weight of a hyperedge is computed by summing the adjacency scores of all the pairs of vertices contained in this hyperedge. The adjacency score of any pair of vertices is defined as the L2 distance of image features:

$$S(u, v) = \exp \left(-\frac{1}{\sigma} \|feat(u), feat(v)\| \right), \quad (12)$$

where $feat(u)$ represents the image feature vector of vertex u . σ is calculated by the the standard deviation of distances. In this way, Ω can be computed by

$$\Omega_u = \sum S(u, v) \text{ where } v \text{ is the } k \text{ nearest neighbors of } u. \quad (13)$$

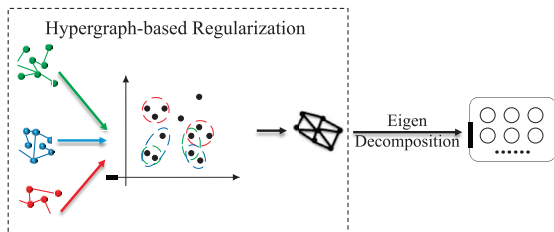


Figure 2 | The process of multiple-feature integration with hypergraph regularization.

Then, DE and DV can be computed by

$$DE_e = \sum_{v=1}^N H(v, e), \quad (14)$$

and

$$DV_v = \sum \Omega_u \text{ where } H(u, e) = 1. \quad (15)$$

With the hyper edge weighting matrix, the multi-feature hypergraph Laplacian can be computed by summing the patch optimization defined in Eq. (10) of all the hyperedges:

$$L = \frac{1}{2} \sum_{e \in E} \sum_{v \in e} \frac{Y}{DV_v^2} H'_e \frac{\Omega}{DE} H_e \frac{Y}{DV_v^2}. \quad (16)$$

The computational complexity of this stage is n^2 .

Then, the overall output can be computed by standard eigen-decomposition on L . The dimensionality of resulting vectors D are determined by the number of eigen values we retain.

3. EXPERIMENTAL EVALUATIONS

3.1. Datasets and Settings

In the experiments, we use two challenging datasets for face spoof attack detection. The first one is the NUAA Photograph Imposter Database (NUAA). NUAA is collected by generic and commonly-used webcams [21]. It is collected in three sessions. The place and illumination conditions of each session are different. There are 5105 real faces and 7509 fake images from 15 subjects in total. Some sample images are shown in Figure 3.

The second dataset is the multispectral-spoof face spoofing database built at Idiap Research institute (MSSPOOF) [22]. It contains both color images and infrared images. Similar to NUAA,

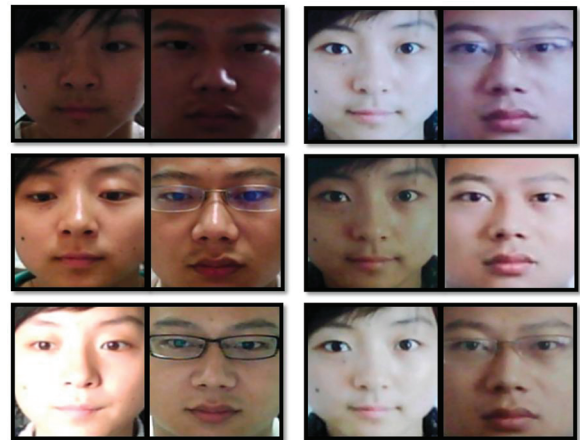


Figure 3 | Sample images in the NUAA dataset are shown. The images in the left two columns are real and the other two columns are fake.



Figure 4 | Sample images in the MSSPOOF dataset are shown. The first row are images taken with in Visible spectra (VIS), while the second one are images taken in Near-Infrared spectra (NIR). The first column are real accesses. The second column are VIS attacks. The third column are NIR attacks.

images in MSSPOOF are recorded in different light conditions. The number of subjects in the database is 21. There are 70 real faces and 144 fake images for each subject. Examples of the database are shown in Figure 4.

In the experiments, the performance is measured by two criteria as error rates. In face spoof attack detection, we pay more attention on the ratio of fake images classified as real faces, which is known as the false negative (FN) rate. However, the ratio of real faces classified as fake images is also used in our experiments, which is known as the false positive (FP) rate. For cross validation, we randomly choose a subset of samples in training and the rest samples are used in testing. The proportion ranges from 10% to 50%. This process is repeated 10 times and the average results are shown. All the facial parts are detected and resized to 200×200 . SVM is used as the classifier for image features. A laptop with i7-9750H CPU, 16G RAM and GTX1650 GPU is used. Evaluations are run on MATLAB R2017a.

3.2. Parameter Sensitivities

There are two parameters to be tuned in the proposed method. They are the number of neighbors and the dimensionality of output features D . To demonstrate the effectiveness of hypergraph regularization, we compare it with existing graph learning methods, such as LDA, DLA, LPP, NPE, LSDA and ISOMAP [20]. For simplicity, only the results on FN rates under 50% training proportion are shown in Figures 5 and 6. Although the proposed hypergraph regularization is not always the best for each number and dimensionality, the overall best performance is still achieved with it. However, for different datasets, the conditions of the best performance are different. For NUAA, the best performance is achieved with $K = 20$ and $D = 150$. On the other hand, for MSSPOOF, the best performance is achieved with $K = 25$ and $D = 200$.

3.3. Comparison with Existing Methods

To demonstrate the performance of face spoof attack detection, the following methods including the proposed HGC-CNN are used in comparison.

- Face anti-spoofing with domain generalization (FAS-DG) [13]. FAS-DG estimates the depth information to get depth loss and

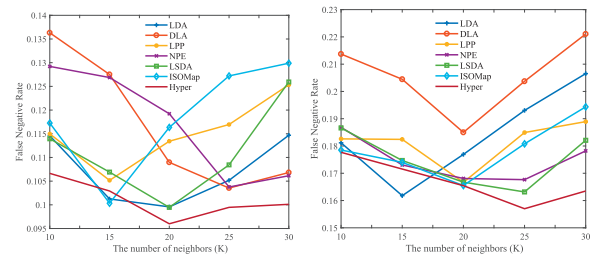


Figure 5 | Comparison under different numbers of neighbors. The left figure is the result of NUAA while the right one is the result of MSSPOOF.

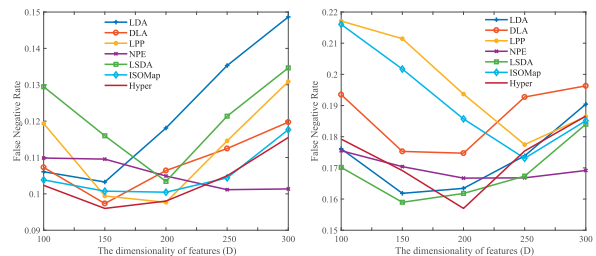


Figure 6 | Comparison under different dimensionality. The left figure is the result of NUAA while the right one is the result of MSSPOOF.

employs a meta-learner to compute classification loss. Both of them are used in face spoof attack detection.

- SeetaFace6. SeetaFace is an open solution for various applications related to faces, such as face detection, face recognition and so on. The latest version, which is named SeetaFace6, provides face spoof attack detection. The model has been trained and we simply use it in testing.
- GS-LBP [7]. GS-LBP makes use of the edge-preserving property of the guided scale space. Besides, joint quantization is used to encode the spatial locality. SVM is used as the classifier.
- HGC-CNNs. The proposed multiple-feature method applied capsule networks and hypergraph regularization.

The results on NUAA is shown in Table 2, while the results on MSSPOOF is shown in Table 3. The best results in each row are highlighted. According to them, we can figure out the following observation.

1. Generally speaking, the performance on NUAA is better than MSSPOOF. Fake Images in MSSPOOF are taken with good quality, which makes it more difficult to distinguish fake images and real faces.
2. FP rates are much lower than FN rates. However, FN rates are more important.
3. The proposed HGC-CNN achieves the best performance on FN rates. However, in some cases, SeetaFace6 achieves the best performance on FP rates.
4. Averagely, HGC-CNN achieves better performance over existing methods.

Table 2 | Comparison on NUAA.

	FAS-DG	SeetaFace6	GS-LBP	HGC-CNN	
*FN	10%	23.6%	22.1%	24.1%	18.1%
	20%	21.9%	21.9%	23.9%	16.3%
	30%	18.3%	22.3%	19.1%	15.0%
	40%	16.5%	23.1%	15.9%	12.4%
	50%	15.1%	21.0%	14.3%	9.6%
*FP	10%	1.2%	0.2%	0.8%	0.9%
	20%	0.9%	0.0%	0.5%	0.5%
	30%	0.8%	0.4%	0.6%	0.4%
	40%	0.4%	0.3%	0.4%	0.2%
	50%	0.2%	0.0%	0.3%	0.0%

Table 3 | Comparison on MSSPOOF.

	FAS-DG	SeetaFace6	GS-LBP	HGC-CNN	
*FN	10%	33.1%	42.9%	29.2%	23.9%
	20%	29.7%	43.1%	26.3%	22.2%
	30%	28.1%	41.9%	25.1%	19.3%
	40%	25.8%	42.1%	23.0%	17.8%
	50%	23.1%	44.3%	19.2%	15.7%
*FP	10%	4.3%	2.9%	5.1%	2.9%
	20%	3.9%	2.7%	4.2%	2.8%
	30%	2.8%	2.5%	3.7%	2.5%
	40%	2.3%	2.3%	2.4%	2.2%
	50%	2.1%	3.4%	3.5%	1.9%

In Tables 2 and 3, the proposed method is not always the best on FP. We have looked into the failed samples of the proposed method. We can figure out that blurred images and dark images may cause failures. Therefore, in the future, some image preprocessing can be introduced to further improve the performance.

4. CONCLUSIONS

In this paper, a novel feature representation for facial images is proposed. It is used in face spoof attack detection with color images and called HGC-CNNs. It is a multiple-feature learning framework combining the ideas of capsule neural networks and hypergraph regularization. Capsule neural networks are able to integrate different types of features, such as pixel intensities, LBP and image quality. Furthermore, hypergraph regularization can be used to learn the correlations among samples. Locality information further improves the descriptive power of output features. The novel representation can be compatible with exiting classifiers and SVM is chosen in the experiments. Experimental results on the NUAA Photogrpah Imposter Database and the Multispectral-Spoof face spoofing database show that the proposed method outperforms previous method on face spoof attack detection with color images.

CONFLICTS OF INTEREST

The authors declare of no conflicts of interest.

AUTHORS' CONTRIBUTIONS

Yuxin Liang: methodology, software, formal analysis and data curation. **Chaoqun Hong:** conceptualization, funding acquisition, investigation, writing–review and editing. **Weiwei Zhuang:** visualization, supervision and validation

ACKNOWLEDGMENTS

This work was partly supported in part by the National Natural Science Foundation of China (No. 61871464 and 61836002), the Fujian Provincial Natural Science Foundation of China (No. 2018J01573), the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (No. VRLAB2021C01) Distinguished Young Scientific Research Talents Plan in Universities of Fujian Province, the Program for New Century Excellent Talents in University of Fujian Province and Xiamen Youth Innovation Fund Project (No. 3502Z20206072).

REFERENCES

- [1] V. Kazemi, J. Sullivan, One millisecond face alignment with an ensemble of regression trees, in IEEE Conference on Computer Vision & Pattern Recognition, Columbus, OH, USA, 2014, pp. 1867–1874.
- [2] A. Anjos, M.M. Chakka, S. Marcel, Motion-based countermeasures to photo attacks in face recognition, IET Biometrics. 3 (2014), 147–158.
- [3] S. Bharadwaj, T.I. Dhamecha, M. Vatsa, R. Singh, Computationally efficient face spoofing detection with motion magnification, in IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Portland, OR, USA, 2013, pp. 105–110.
- [4] J. Liu, A. Kumar, Detecting presentation attacks from 3d face masks under multispectral imaging, in IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Salt Lake City, UT, USA, 2018, pp. 47–52.
- [5] Y. Wang, F. Nian, T. Li, Z. Meng, K. Wang, Robust face anti-spoofing with depth information, J. Vis. Commun. Image Represent. 49 (2017), 332–337.
- [6] R. Bresan, A. Pinto, A. Rocha, C. Beluzo, T. Carvalho, Facespoofer: a presentation attack detector based on intrinsic image properties and deep learning, Pattern Recognit. Lett. (2019). https://www.researchgate.net/publication/331008526_FaceSpoofer_Buster_a_Presentation_Attack_Detector_Based_on_Intrinsic_Image_Properties_and_Deep_Learning
- [7] F. Peng, L. Qin, M. Long, Face presentation attack detection using guided scale texture, Multimedia Tools Appl. 77 (2018), 8883–8909.
- [8] J. Galbally, S. Marcel, Face anti-spoofing based on general image quality assessment, in International Conference on Pattern Recognition, IEEE, Stockholm, Sweden, 2014, pp. 1173–1178.
- [9] J. Yang, Z. Lei, S.Z. Li, Learn convolutional neural network for face anti-spoofing, arXiv: Computer Vision and Pattern Recognition, 2014.
- [10] L. Feng, L.-M. Po, Y. Li, X. Xu, F. Yuan, T.C.-H. Cheung, K.-W. Cheung, Integration of image quality and motion cues for

- face anti-spoofing, *J. Vis. Commun. Image Represent.* 38 (2016), 451–460.
- [11] L. Lei, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, A. Hadid, An original face anti-spoofing approach using partial convolutional neural network, in *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Oulu, Finland, 2017.
- [12] Y. Liu, A. Jourabloo, X. Liu, Learning deep models for face anti-spoofing: binary or auxiliary supervision, in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, UT, USA, 2018, pp. 389–398.
- [13] R. Shao, X. Lan, P.C. Yuen, Regularized fine-grained meta face anti-spoofing, *arXiv: Computer Vision and Pattern Recognition*, 2019.
- [14] J. Yu, M. Tan, H. Zhang, D. Tao, Y. Rui, Hierarchical deep click feature prediction for fine-grained image recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* (2019), 1. <https://ieeexplore.ieee.org/abstract/document/8781933>
- [15] J. Yu, Z. Kuang, B. Zhang, W. Zhang, D. Lin, J. Fan, Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing, *IEEE Trans. Inf. Forensics Secur.* 13 (2018), 1317–1332.
- [16] J. Yu, C. Zhu, J. Zhang, Q. Huang, D. Tao, Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition, *IEEE Trans. Neural Netw.* 31 (2020), 661–674.
- [17] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in *International Conference on Learning Representations*, Toulon, France, 2017.
- [18] S. Sabour, N. Frosst, G.E. Hinton, Dynamic routing between capsules, in: *Advances in Neural Information Processing Systems*, MIT Press, Long Beach, USA, 2017, pp. 3856–3866.
- [19] C. Hong, J. Yu, J. Li, X. Chen, Multi-view hypergraph learning by patch alignment framework, *Neurocomputing.* 118 (2013), 79–86.
- [20] T. Zhang, D. Tao, X. Li, J. Yang, Patch alignment for dimensionality reduction, *IEEE Trans. Knowl. Data Eng.* 21 (2009), 1299–1313.
- [21] X. Tan, L. Yi, J. Liu, J. Lin, Face liveness detection from a single image with sparse low rank bilinear discriminative model, in *European Conference on Computer Vision: Part VI*, Heraklion, Greece, 2010, pp. 504–517.
- [22] A. Anjos, S. Marcel, I. Chingovska, N. Erdogmus, Face recognition systems under spoofing attacks, in: T. Bourlai (Ed.), *Face Recognition Across the Imaging Spectrum*, Springer, Cham, Switzerland, 2015.