Spatio-Temporal Texture Features for Presentation Attack Detection in Biometric Systems

Shi Pan
University of Kent,
Canterbury Kent
CT2 7NT
Kent, UK
sp641@kent.ac.uk

Farzin Deravi
University of Kent,
Canterbury Kent
CT2 7NT
Kent, UK
f.deravi@kent.ac.uk

***Abstract*—Spatio-temporal information is valuable as a discriminative cue for presentation attack detection, where the temporal texture changes and fine-grained motions (such as eye blinking) can be indicative of some types of spoofing attacks. In this paper, we propose a novel spatio-temporal feature, based on motion history, which can offer an efficient way to encapsulate temporal texture changes. Patterns of motion history are used as primary features followed by secondary feature extraction using Local Binary Patterns and Convolutional Neural Networks, and evaluated using the Replay Attack and CASIA-FASD datasets, demonstrating the effectiveness of the proposed approach.**

Keywords—Presentation Attack Detection, Facial Biometrics, Local Binary Patterns, Convolutional Neural Networks, Spatio-temporal Features

I. INTRODUCTION

Face recognition is a powerful solution for authentication systems and has consistently attracted the attention of both academic and industrial researchers for its further improvement. Recent developments in deep learning techniques has greatly enhanced the performance of automatic face recognition systems even in unconstrained environments [1]. However, zero-effort facial spoofing or presentation attacks (PA) has put additional obstacles in the path of the wider adoption of facial biometric systems [2]. Facial presentation attacks, by presenting a faked biometric evidence from a valid user at the sensor, create a serious challenge to the trust in biometric systems as they only require that attackers collect some biometric samples from valid users to construct an attack artefact. For instance, some face recognition system can be simply attacked by presenting the printed facial information of a valid user. Moreover, the popularity of social media (such as Facebook) highly reduces the cost of acquiring high-resolution facial biometric samples.

The urgent need for protecting face recognition systems from such attacks has stimulated the development of presentation attack detection (PAD) systems. For face recognition systems, presentation attacks can be categorized by the attack artefacts, including printed papers (paper attack), screens (video replay attack), and 3D masks (mask attack) [3]. Here, the valid biometric samples are known as the *bona-fide* class (of genuine/real samples). For developing and evaluating robust PAD algorithms, various public datasets, including various attack types and environmental conditions, are used.

In general, presentation attacks can be recognised by human observers via the material differences between genuine faces and attack artefacts; the different representations between the non-rigid facial movements and rigid movements for artifacts; and the texture differences between the recaptured images and original images. Thus, some researchers in this area aim to build software-based methods which can detect attacks without costly additional hardware. This research direction is known as software-based PAD or feature-based PAD [3]. One of the key assumptions of current feature-based PAD researches is that attacks can be detected by the distorted information which is injected into the sensor data during the spoofing attack by the martial of attack artefacts, changing both the spatial and temporal appearance of the data when compared with a bona-fide presentation. PAD research, therefore, has explored both static and dynamic feature-based methods [4], where the static methods are only aimed at detecting the traces of artefacts in the spatial domain and the dynamic methods also aim at such detection in the temporal domain. Dynamic feature-based methods have also been explored in the past, such as using texture differences [5], motion differences [6] and image quality differences [7] for PAD.

In this context, the proposed work aims to provide a novel discriminative spatio-temporal feature for presentation attack detection capturing the texture changes in a frame sequence. The temporal changes are compressed and represented by a single image using the Motion History Image algorithm [19]. The texture patterns inn the spatio-temporal domain are modelled by using Local Binary Patterns (LBP) [11] and Convolutional Neural Networks (CNN) [15]. The proposed work offers a new direction to produce PAD-related information by efficiently encapsulating the distinct spatio-temporal information to texture patterns and significantly decreasing the computational effort required for modelling dynamic textures. To demonstrate the effectiveness of this spatio-temporal feature, the efficiency of the proposed approach is evaluated by testing it on two widely-used datasets.

II. Relate work

The existing PAD research includes various research directions such as static texture analysis [8], dynamic texture analysis [5] and challenge-responses strategies [9]. Here, we only consider the relevant works which aim at detecting facial presentation attacks by exploring spatio-temporal texture changes.

The related work for detecting presentation attacks via the spatio-temporal texture change may be divided into two categories: work using Deep Neural Networks (DNNs) or not using DNNs; the latter being also known as traditional features or hand-crafted features [1]. LBPs are established features for PAD in the traditional feature category. They have been shown to perform well for some datasets. Although, other texture descriptors (such as the Haralick texture descriptor [10]) are effective for some attack categories, LBPs demonstrate efficiency and robustness for most attack categories and evaluation datasets. Moreover, LBPs are considered as the baseline algorithm for many evaluation schemes [11] due to their computational efficiency.

Various extensions of LBPs have been applied for spoofing attack detection since LBPs were firstly explored by Ojala et al. [12] including the combination of LBPs (such as colour LBP [8] and multi-scale LBP[13]). For the spatio-temporal texture changes, one of the most famous extensions for LBP is LBP-TOP. In LBP-TOP, Pereira et al. [14] calculate LBP descriptors for three selected orthogonal planes (X-Y, X-T, and Y-T) to explore temporal data. It compresses the desired time-related information into texture changes of two orthogonal planes. However, in the implementation of [14], two selected time-related orthogonal planes (X-T, and Y-T) discard much texture changes which may be related with presentation attack. This disadvantage inspired the proposed work for a robust spatio-temporal feature for PAD.

On the other hand, the methods using DNNs also aim to model the spatio-temporal changes for PAD. One of the well-known methods using DNNs is combining a convolutional neural network (CNN) and Long Short Term Memory (LSTM) for modelling the desired spatio-temporal changes. The success of CNNs in many pattern recognition applications has established the effectiveness of a well-designed deep convolutional architecture [23]. And the popularity of the transfer learning paradigm for pre-trained CNNs has pushed performance boundaries in various areas. Some researchers have introduce this paradigm into the PAD area and obtained promising results [15]. Also, the effectiveness of LSTM for modelling time-related information has been demonstrated in various areas. Z, Xu et al. [16] proposed a CNN+LSTM architecture for PAD and demonstrated competitive performance for some datasets. Other methods such as CNN LBP-TOP [17] which inherit ideas from traditional features but also use DNNs to model spatio-temporal information, also show promise of enhanced performance.

III. Motivation

In this paper, we propose a novel time-based PAD algorithm by combining Motion History Image (MHI) as primary features and two local texture descriptors as secondary features. The PAD related temporal changes can be identified with some local texture descriptors by establishing an algorithm which can transfer temporal differences to texture patterns. Some previous works (such as LBP-TOP and texture co-occurrence patterns) has demonstrated the feasibility of this general approach. The proposed work develops this idea to model temporal changes but also explores different ways to create time-related texture difference patterns.

Moreover, the proposed method is also focusing on exploring temporal texture differences and local texture co-relations between frames. For instance, in [4], the moiré pattern is considered as a significant indicator for detecting video attacks. However, moiré patterns may not be visible in every frame of the video. Some evaluation datasets, even with higher video quality, may still contain frames where moiré patterns are not visible. The disappearance of the moiré patterns makes the modelling of temporal local textures difficult, especially for shorter presentations (video sequences). However, these temporal texture changes (such as the appearance and disappearance of moiré patterns) can be easily enhanced and identified by using the frame difference method [18]. Furthermore, these temporal texture differences for PAD could appear in almost any location within a frame. Inspired by these facts, the proposed method is focusing on exploring temporal texture changes and transferring these changes into spatial texture patterns.

The frame difference algorithm can only represent texture changes between two selected frames. However, not all the desired dynamic texture changes will appear between two selected frames. And applying the frame difference algorithm for each frame will produce a frame difference image sequence, thus enlarging the volume of data that needs to be processed. Furthermore, the texture changes between two frames may not be significant enough for PAD as in many cases the frame difference will not include significant temporal texture changes (such as moiré patterns). Also, object movements (such as body movements and facial movements) will also represent large pixel value changes that are not necessarily significant for PAD.

To overcome the limitations of the frame difference algorithm, the Motion History Image (MHI) is introduced to provide primary features for PAD-related temporal texture changes which are combined with a local texture descriptor (such as the LBP) to produce secondary features for PAD. According to Bobick and Davis [19], object movements can be decomposed using MHI by describing where the motion appeared and how the object moves. In this way the texture changes caused by object movements can be collected for recognition. One of the advantages of their idea is that the desired object movements and texture changes may be compressed and encoded into the spatial texture changes within a single frame.

There are two steps to produce an MHI. In the first step, the binary Motion Energy Image (MEI) is created and transformed into a Binary Motion Region (BMR) mask to represent the spatial relationship for the motions that have occurred in the image sequence. These BMR masks encapsulate temporal texture changes which have different characteristics for different presentation attack categories. For instance, paper attacks may

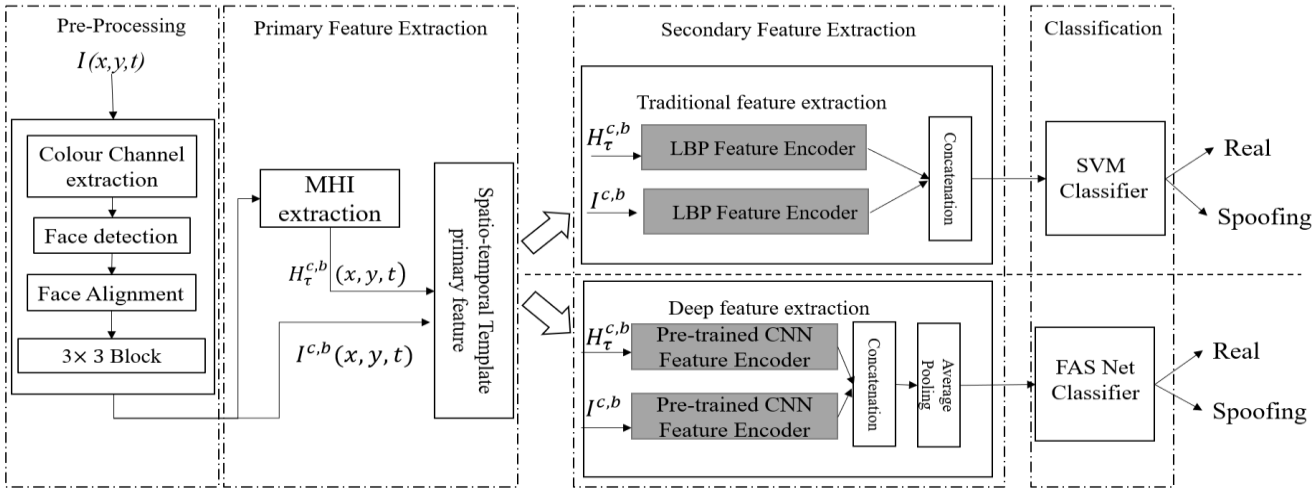

Figure 1. Experimental workflow

include significant texture changes caused by different movement trajectories between faces and attack artefacts. These trajectories will be represented by different spatial locations in the sequence of MEI. The motion regions in MEI include the information about the motion-shapes and the spatial distribution of motions. The particular shapes of the motion texture patterns such as moiré pattern will be enhanced in the MEI.

In the second step, the BMR mask sequence is compressed to generate the MHI by calculating a function of motion density at each pixel location. The intensity value of each pixel is a function of the motion at that pixel position. The original MHI algorithm can only be applied for fixed cameras. The data from hand-held cameras would need an optical flow algorithm as an additional pre-processing step.

The proposed spatio-temporal feature construction consists of two parts: (1) The spatial component of the feature is normally the first image in the frame sequence which is used as the first image for calculating the MHI. (2) The temporal component of the proposed feature is the MHI itself.

IV. METHODOLOGY

The overall workflow of the proposed experiments exploring two different secondary feature extractors is presented in Fig. 1. For each frame sequence multiple colour channels (HSV and YCbCr) are extracted and the algorithms will be applied on all of these colour channels. Then, the proposed method detects the facial area and performs face alignment by using eye positions. After that, the cropped facial area is divided into 3x3 blocks. For each of the blocks at each colour channel, a block sequence is formed for calculating MHI. This sequence, together with the first frame of the video are used as the primary spatio-temporal feature. Then, the local texture descriptors for the spatial texture and motion history texture are calculated separately for each of the block sequences. The final feature vector is constructed by the concatenation of multiple local texture descriptors. The proposed experimental workflow considers two secondary feature extractors separately and uses two different classifiers for different feature extractors. The system consisting of the LBP as the secondary feature extractor and Support Vector Machine (SVM) classifier is a traditional classification approach. The alternative system consisting of a pre-trained CNN and a FAS Net classifier has elements of a deep learning approach.

The Motion History Image (MHI) [19], $H_\tau(x,y,t)$ can be calculated by using an update function $\Psi(x,y,t)$

$$H_\tau(x,y,t) = \begin{cases} \tau & if\ \Psi(x,y,t) = 1 \\ \max(0, H_\tau(x,y,t-1) - \delta) & otherwise \end{cases} \quad (1)$$

Where (x, y) represent the spatial location of movement and t show the time point. $\Psi(x,y,t)$ is an indicator function to represents whether important temporal texture changes (moiré patterns) or object movements (e.g. head motion) are present in the current video frame. The temporal extent of the texture changes and movements is represented by the duration τ. The δ denotes the decay, which is used to make reduces the influence of earlier texture. Each new video frame will call this update function to calculate the correlated Motion History Image as the temporal feature. The indicator function $\Psi(x,y,t)$ is calculated from a binarized frame difference image using a threshold ξ:

$$\Psi(x,y,t) = \begin{cases} 1 & if\ D(x,y,t) \geq \xi \\ 0 & otherwise \end{cases} \quad (2)$$

Where $D(x,y,t) = |I(x,y,t) - I(x,y,t \pm \Delta)|$ And $I(x,y,t)$ is the intensity value of pixel location with coordinate spatial location (x, y) at the t-th frame of the image sequence. When the desired MHI is extracted from a frame sequence, the local texture descriptors are calculated for PAD. Then, the MHI and original frame are used together to form the spatio-temporal feature as described below.

In order to show the discriminative capability of the proposed spatio-temporal feature, two widely-used algorithms (LBP and CNN) for texture feature processing are are considered as the secondary feature in the proposed workflow for PAD.

In the system using traditional feature extraction method, the proposed workflow considers LBP as the secondary feature to describe texture patterns in the original frame and the MHI. The LBPs are widely used [11] for PAD as a highly discriminative texture descriptor for the proposed spatio-temporal feature. For each pixel in the proposed feature, LBP can be defined with (3) and (4) by following [12] when given a central pixel at (x_{cen}, y_{cen}) and using g_p,. g_c to represents grey-level values.

$$LBP_{P,R} = \Sigma_{p=0}^{P-1} s(g_p - g_c) * 2^p \quad (3)$$

$$s(Z) = \begin{cases} 1 & if\ Z > 0 \\ 0 & otherwise \end{cases} \quad (4)$$

$$U(LBP_{P,R}) = \Sigma_{p=1}^{P-1} |Sig(g_{p-1} - g_c) - Sig(g_p - g_c)| + |Sig(g_{P-1} - g_c) - Sig(g_0 - g_c)| \quad (5)$$

where P is the number of sampling points in a circular neighbourhood set of radius R centred at (x,y); and g_p indicates the pixel value of p-th point on this neighbourhood. The pixel value of the central points g_c is used as a threshold for g_p. In order to reduce the number of valid LBP codes and reduce the dimension of the LBP descriptor, the Uniform function in (5) is integrated into the LBP [11], where the Uniform LBP only considers the binary patterns which $U(LBP_{P,R})$ <2. Various unified LBP descriptors for different colour channels and blocks are concatenated to construct the final feature vector.

In the alternative system using convolutional neural network to extract the secondary feature, the proposed method uses a pre-trained CNN to generate feature descriptors. The convolutional neural network is a well-known algorithm for texture feature extraction. As shown by Lucena et al. [15], applying a pre-trained CNN using the transfer learning paradigm can improve the robustness of features and avoids the overfitting problem for PAD. Thus, the proposed workflow also uses the feature extraction part of a pre-trained CNN. The full architecture of the selected pre-trained DNNs should be designed for large-scale image classification problems. The efficiency of such deep neural architectures is demonstrated by the competitive performance score for the image classification competitions such as the ImageNet competition [20]. The transfer learning paradigm reuses the feature extraction part of the pre-trained network and trains a new classifier on the target domain by transferring the expressive feature space which is learned by the pre-trained neural network [25].

The transfer learning paradigm of a pre-trained deep neural architecture can be defined as the following steps: (1) The pre-trained DNNs, which include the deep neural architectures and all of the parameters in the networks, should be divided into feature extractor parts and classification parts by following [15]. The original classification layers should be replaced by a new classification sub-network. This new classification sub-network is initialised and trained for presentation attack detection. (2) The new network with pre-trained feature extraction part and replaced classification part is trained using presentation attack datasets with different learning rates. The classification sub-network can follow the suggested learning rate of [15]. But the pre-trained feature extraction part should start with the lower learning rate than the classification sub-network. (3) The whole network is fine-tuned with a low learning rate to get better performance.

In the proposed alternative system using CNN to produce the secondary feature, the initial frame of the video sequence and the related MHI are fed into the feature extractor network. Then, the feature vectors from the frame and the related MHI are concatenated as the feature descriptor for the proposed spatio-temporal feature. One average pooling layer is applied to combine various colour channels and image blocks to generate the feature descriptor. And this final feature vector is then fed into the classification sub-network.

V. Experiment and Implementation Detials

To demonstrate the effectiveness of the proposed method, two presentation attack detection datasets are considered in the experiments: the CASIA-FASD dataset [11] and the Replay-Attack dataset [22]. These two datasets are widely used datasets for presentation attack detection, which contain several recordings of the genuine client accesses and recordings of various spoofing attack attempts. Some published traditional feature-based PAD methods and DNN-based PAD methods are trained and evaluated with these two datasets. They offer a fair comparison with the state-of-the-art approaches to show the effectiveness of the proposed method.

The CASIA-FA database [11] consists of 600 video clips which include both real and spoofing access attempts, totally, there are 50 individuals listed in the dataset, where the spoofing artefacts are produced from high-quality records of genuine faces. Three different attack artefacts are included: warped photo attacks, cut photo attacks, and video attacks. All of them are designed to simulate real attack attempts. For instance, the cut photo attack is a special photo attack, in which a high-quality face is printed on paper, but where the area surrounding the eyes is cut to subvert eye-motion-based spoofing attack detection methods. Three different image resolutions are used in this dataset to simulate different usage conditions, namely low resolution, normal resolution, and high resolution. In their evaluation scenarios, 50 subjects are split into two categories: the training set (20 subjects) and the test set (30 subjects). They also designed seven detailed scenarios which are: (1) low-quality, (2) normal-quality (3) high-quality, (4) warped photo attacks, (5) cut photo attacks, and (6) video attacks. The (1), (2), and (3) scenarios are used to test the robustness at different image quality conditions. The (4), (5), and (6) scenarios are used to simulate different attack behaviours. The overall test scenario (7) provides combined performance test results for all attack types and qualities.

The Replay-Attack database [22] is another widely used face spoofing dataset which contains various attack behaviours and contains 1300 video clips. There are 50 clients recorded for both real access attempts and 3 different attack behaviours. Two illumination conditions are considered: controlled and adverse. For each condition, three attack categories are included: (1) print attacks, (2) mobile attacks, and (3) high-definition attacks. The mobile attacks and high-def attacks can both be categorised as video attacks but use different sizes of the screen with different resolutions. They also consider various conditions about whether the attack device is fixed in front of the camera: (1) hand based attack (the attack devices were held by hand) and (2)

TABLE I COMPARISON WITH THE STATE-OF-THE-ART LBP-BASED PAD METHODS ON CASIA-FASD AND REPLAY-ATTACK DB OVERALL TEST

	CASIA-FA (EER)	***Replay-Attack DB (HTER)***
baseline LBP[11]	25.0	13.7
Colour LBP [8]	2.1	3.5
LBP-TOP[14]	10.6	7.6
Proposed MHI-LBP	4.8	3.9

fixed-support attacks (the attack devices were fixed on a stand). The Replay-Attack database divides the whole dataset into three subsets, which are: the training set, the development set, and the testing set.

The pre-processing for the proposed workflow is important. The fusion of multiple colour spaces is a commonly used approach to enhance performance. The proposed method follows Boulkenafet et al's work [8] to consider multiple colour spaces to explore the colour texture information for PAD. The reason behind this is that the character of the artefact may be more visible in the local uniform areas (e.g. cheeks). . For this reason, the proposed method crops the facial area into 3×3 patches after face alignment and face normalisation. The final feature vector is the concatenation of multiple colour channels and all of the cropped patches.

In the proposed system using LBP to produce secondary feature, a Support Vector Machine (SVM) with the RBF kernel is used as the classifier for comparison with other works using the CASIA-FASD and Replay-Attack datasets. The alternative system using CNN includes two important factors: the pre-trained feature extractor network and the classifier network.

The selection of the pre-trained feature extractor network is important for the proposed alternative system using CNN as the secondary feature. In this system, the pre-trained VGG16 [21] network is considered as a texture feature extractor network in our implementation. The original VGG 16 network, which includes 16 convolutional layers with 3×3 kernel size, is a 2D convolutional neural network for the ImageNet competition [20]. They use rectifier linear unit (ReLU) as activation function and 3 dense layers (or fully connected layers (FC)) for classification. The original dense layers are removed for transfer learning. The classifier network is another important factor for the proposed method. The proposed method follows Lucena et al. [15]'s suggestion which uses a new classification sub-network consisting of one flattened layer, one dense layer with ReLU activation, one dropout layer [24], and one dense layer with a sigmoid activation function. In the training stage, the classification sub-network is optimized by using the initial learning rate of10^{-4}. The pre-trained feature extractor network is optimized by using the initial learning rate of 10^{-6}. The last dense layer uses a sigmoid activation function.

The experiments to evaluate the performance of the proposed method use a four-fold subject-disjoint cross-validation using the CASIA-FA training set due to the absence of a development subset for this dataset. The performance is reported by using the Equal Error Rate (EER) on the test set [11]. The evaluation protocols of the Replay-Attack database require producing the EER on the development set and the Half Total Error Rate (HTER) on the test set [22].

TABLE II COMPARISON WITH THE STATE-OF-THE-ART DNN-BASED PAD METHODS ON CASIA-FASD AND REPLAY-ATTACK DB OVERALL TEST (THE * MEANS THE PERFORMANCE SCORE FROM OUR IMPLEMENTATION FOLLOWING THE REFERENCED WORK)

	CASIA-FA (EER)	***Replay-Attack DB (HTER)***
CNN[23]	6.2	2.6
FASNet*[15]	8.6*	3.9*
CNN+LSTM [16]	5.8*	6.3*
CNN- LBP-TOP [17]	8.0	4.7
Proposed MHI-CNN	6.0	4.5

VI. RESULTS AND ANALYSIS

The proposed method is compared with 7 other approaches which include baseline LBP [11], Colour LBP [8], LBP-TOP[14], CNN [23], FASNet [15], CNN+LSTM [16], and CNN- LBP-TOP [17]. We firstly compared the LBP-based methods in Table 1. The implementation detail of baseline LBP has followed the CASIA-FASD protocol. Then, Colour LBP can be considered as the representative method of static feature-based PAD algorithms, The LBP-TOP is also designed for the spatio-temporal texture changes which use LBP as the texture descriptor. From the Table I, the proposed MHI-LBP is seen to provide good performance scores for both datasets. Although the Colour LBP as a static-texture method represents better performance than the proposed method, the proposed MHI-LBP shows a better performance when compared with LBP-TOP using LBP as the texture descriptor.

The, we compared the CNN-based methods at Table II. The CNN [23] is the first published work using convolutional neural network for PAD. The FASNet also uses a pre-trained VGG16 as the feature encoder and use the transfer learning paradigm to fine-tuning their networks. It demonstrates the effectiveness of the proposed feature. The original FASNet do not train and test their algorithm on the CASIA-FA dataset. We follow their paper and re-implement their algorithm for the CASIA-FA dataset. The CNN+LSTM method represents the effectiveness of the end-to-end neural network for spatio-temporal texture changes. We also re-implement their work for the comparison on the Replay-Attack Dataset. The CNN-LBP-TOP as a hybrid method which combines traditional features and DNNs are also considered for the comparison.

The proposed MHI-CNN provides the best performance for the CASIA-FA dataset when compared to the listed CNN-based methods. It demonstrates the effectiveness of the proposed spatio-temporal feature. On the Replay Attack Dataset, CNN [23] showed the best performance. However, it includes multiple data augmentation stages and is trained from scratch.

Some have claimed that these are overfitting their training data [17]. The proposed method uses the pre-trained CNN architecture to overcome the overfitting problem. Moreover, the proposed spatio-temporal feature outperforms those proposed in [16] and [17]'s when evaluated on the Replay-Attack dataset.

VII. Conclusion

In this paper, novel spatio-temporal feature combinations are explored which are based on motion history images as primary features. The proposed MHI-texture descriptor constructs MHIs as a description of the temporal texture changes and uses LBP and CNN to produce secondary feature vectors. The MHI-LBP combination offers a method to encode temporal texture changes with low computational complexity. The MHI-CNN combination uses a pre-trained CNN and the transfer learning paradigm as a new hybrid framework which uses both traditional features and deep neural networks for PAD. These combined features when evaluated using two widely used datasets demonstrate a good performance compared with some similar state-of-the-art techniques.

References.

[1] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in British Machine Vision Conference, vol. 1. BMVA Press, 09 2015, pp. 41.1 – 41.12.

[2] Ramachandra, Raghavendra, and C Busch. "Presentation attack detection methods for face recognition systems: a comprehensive survey." ACM Computing Surveys (CSUR) 50.1 (2017): 8.

[3] Komulainen, Jukka, Zinelabidine Boulkenafet, and Zahid Akhtar. "Review of Face Presentation Attack Detection Competitions." Handbook of Biometric Anti-Spoofing. Springer, Cham, 2019. 291-317.

[4] Galbally, Javier, Sébastien Marcel, and Julian Fierrez. "Biometric antispoofing methods: A survey in face recognition." IEEE Access 2 (2014): 1530-1552.

[5] S Pan, and F Deravi. "Facial biometric presentation attack detection using temporal texture co-occurrence." 2018 IEEE 4th International Conference on Identity, Security, and Behavior Analysis (ISBA). IEEE, 2018.

[6] K. Kollreider, H. Fronthaler, M. I. Faraj, and J. Bigun, "Real-time face detection and motion analysis with application in 'liveness' assessment," IEEE Trans. Inf. Forensics Security, vol. 2, no. 3, pp. 548–558, Sep. 2007.

[7] J. Galbally, S. Marcel, and J. Fierrez, "Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition," IEEE Trans. Image Process., vol. 23, no. 2, pp. 710–724, Feb. 2014.

[8] Boulkenafet, Zinelabidine, Jukka Komulainen, and Abdenour Hadid. "Face spoofing detection using colour texture analysis." IEEE Transactions on Information Forensics and Security 11.8 (2016): 1818-1830.

[9] De Marsico, Maria,. "Moving face spoofing detection via 3D projective invariants." 2012 5th IAPR International Conference on Biometrics (ICB). IEEE, 2012.

[10] Agarwal, Akshay, Richa Singh, and Mayank Vatsa. "Face anti-spoofing using Haralick features." 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS). IEEE, 2016.

[11] Zhiwei Zhang, Junjie Yan, Sifei Liu, Zhen Lei, Dong Yi, and S.Z. Li, "A face antispoofing database with diverse attacks," in International Conference on Biometrics (ICB), March 2012, pp. 26–31

[12] Ojala, Timo, Matti Pietikäinen, and Topi Mäenpää. "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns." IEEE Transactions on Pattern Analysis & Machine Intelligence 7 (2002): 971-987.

[13] Määttä, Jukka, Abdenour Hadid, and Matti Pietikäinen. "Face spoofing detection from single images using micro-texture analysis." 2011 international joint conference on Biometrics (IJCB). IEEE, 2011.

[14] de Freitas Pereira, Tiago, et al. "LBP− TOP based countermeasure against face spoofing attacks." Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.

[15] Lucena, O., Junior, A., Moia, V., Souza, R., Valle, E., & Lotufo, R.. "Transfer learning using convolutional neural networks for face anti-spoofing." International Conference Image Analysis and Recognition. Springer, Cham, 2017.

[16] Xu, Zhenqi, Shan Li, and Weihong Deng. "Learning temporal features using LSTM-CNN architecture for face anti-spoofing." 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR). IEEE, 2015.

[17] Asim, Muhammad, Zhu Ming, and Muhammad Yaqoob Javed. "CNN based spatio-temporal feature extraction for face anti-spoofing." 2017 2nd International Conference on Image, Vision and Computing (ICIVC). IEEE, 2017.

[18] Singla, Nishu. "Motion detection based on frame difference method." International Journal of Information & Computation Technology 4.15 (2014): 1559-1565.

[19] Ahad, M. A. R., Tan, J. K., Kim, H., & Ishikawa, S. "Motion history image: its variants and applications." Machine Vision and Applications 23.2 (2012): 255-281.

[20] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.

[21] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[22] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in International Conference of the Biometrics Special Interest Group (BIOSIG), Sept 2012, pp. 1–7. 3, 4

[23] J. Yang, Z. Lei, and S. S. Z. Li, "Learn Convolutional Neural Network for Face Anti-Spoofing," ArXiv Prepr., no. 1408-5601, p. 8, 2014.

[24] Baldi, Pierre, and Peter J. Sadowski. "Understanding dropout." *Advances in neural information processing systems*. 2013.

[25] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. "A survey on deep transfer learning." International Conference on Artificial Neural Networks. Springer, Cham, 2018.

(references)