Face Anti-Spoofing Using Patch and Depth-Based CNNs

Conference Paper · October 2017

DOI: 10.1109/BTAS.2017.8272713

Yaojie Liu
Google Inc.
19 PUBLICATIONS **945** CITATIONS

Amin Jourabloo
Michigan State University
18 PUBLICATIONS **1,520** CITATIONS

Xiaoming Liu
Michigan State University
221 PUBLICATIONS **13,021** CITATIONS

Face Anti-Spoofing Using Patch and Depth-Based CNNs

Yousef Atoum* Yaojie Liu* Amin Jourabloo* Xiaoming Liu
Department of Computer Science and Engineering
Michigan State University, East Lansing MI 48824
{atoumyou, liuyaoj1, jourablo, liuxm}@msu.edu

Abstract

The face image is the most accessible biometric modality which is used for highly accurate face recognition systems, while it is vulnerable to many different types of presentation attacks. Face anti-spoofing is a very critical step before feeding the face image to biometric systems. In this paper, we propose a novel two-stream CNN-based approach for face anti-spoofing, by extracting the local features and holistic depth maps from the face images. The local features facilitate CNN to discriminate the spoof patches independent of the spatial face areas. On the other hand, holistic depth map examine whether the input image has a face-like depth. Extensive experiments are conducted on the challenging databases (CASIA-FASD, MSU-USSA, and Replay Attack), with comparison to the state of the art.

1. Introduction

Biometrics utilize physiological, such as fingerprint, face, and iris, or behavioral characteristics, such as typing rhythm and gait, to uniquely identify or authenticate an individual. As biometric systems are widely used in real-world applications including mobile phone authentication and access control, biometric spoof, or Presentation Attack (PA) are becoming a larger threat, where a spoofed biometric sample is presented to the biometric system and attempted to be authenticated. Since face is the most accessible biometric modality, there have been many different types of PAs for faces including print attack, replay attack, 3D masks, etc. As a result, conventional face recognition systems can be very vulnerable to such PAs.

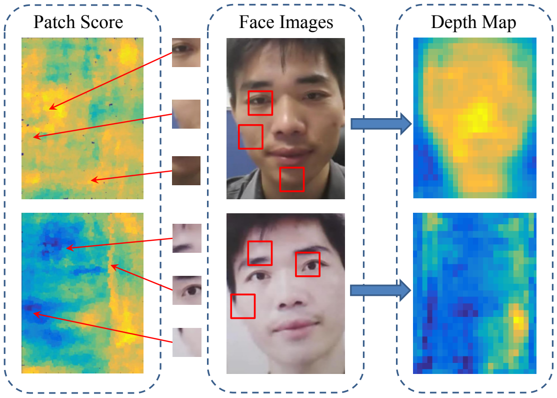

Figure 1: In order to differentiate between live from spoof images, we propose an approach fusing patch-based and holistic depth-based cues. Left column shows the output scores of the local patches for a live image (top) and a spoof image (bottom), where the blue/yellow represent a high/low probability of spoof. While this visualization utilizes densely sampled patches, 10 random patches are sufficient for our anti-spoof classification. Right column shows the output of holistic depth estimation, where the yellow/blue represent a closer/further points.

In order to develop a face recognition system that is invulnerable to various types of PAs, there is an increasing demand on designing a robust face anti-spoofing (or PA detection) system to classify a face sample as live or spoof *before* recognizing its identity [3, 4]. Previous approaches to tackle face anti-spoofing can be categorized in three groups. The first is the texture-based methods, which discover discriminative texture characteristics unique to various attack mediums. Due to a lack of an explicit correlation between pixel intensities and different types of attacks, extracting robust texture features is challenging. The second is the motion-based methods that aim at classifying face videos based on detecting movements of facial parts, e.g., eye blinking and lip movements. These methods are suitable for static attacks, but not dynamic attacks such as replay or mask attacks. The third is image quality and reflectance-based methods, which design features to capture the superimposed illumination and noise information to the spoof images.

Most of the prior face anti-spoofing work, as one of our key observations, apply a classifier on hand-crafted features. While Convolutional Neural Network (CNN) exhibits its superior performance in many computer vision tasks [26, 27, 22], there are only a few CNN-based meth-

*denotes equal contribution by the authors.

ods for face anti-spoofing. Existing CNN methods typically use CNN for learning representations, which will be further classified by SVM [29, 33]. In our view, further utilizing CNN in multiple ways, such as end-to-end training and learning with additional supervision, is a viable option for solving face anti-spoofing problems. On one hand, with an increasing variety of sensing environments and PAs, it is not desirable to have a hand-crafted feature to cover all attacks. On the other hand, we need CNN to learn a robust feature from the data. With the growing numbers of face spoofing databases, CNN is known to be able to leverage the larger amount of training data, and learn generalizable information to discriminate live vs. spoof samples.

Following this perspective, as shown in Figure 1, this paper proposes a novel two-stream CNN-based face anti-spoofing method, for print and replay attacks. The proposed method extracts the local features and holistic depth maps from face images. Here the local features are extracted from random patches within the face region, while the depth features leverage the whole face, and describe the live face as a 3D object but the spoof face as a flat plain (assuming PAs include print attack and replay attack). Since face spoofing datasets contain videos with different qualities, combining the local and holistic features has two benefits: First, utilizing the local patches help to learn spoof patterns independent of spatial face areas. Second, holistic depth maps ensure the input live sample has a face-like depth. Hence, we use two CNNs to learn local and holistic features respectively. The first CNN is end-to-end trained, and assign a score to each randomly extracted patch from a face image. We assign the face image with the average of scores. The second CNN estimates the depth map of the face image and provide the face image with a liveness score based on estimated depth map. The fusion of the scores of both CNNs lead to the final estimated class of live vs. spoof.

We summarize our main contributions as follows:

◇ Our proposed method utilizes both learned local and holistic features for classifying live vs. spoof face samples.

◇ We propose a method for estimating the dense depth map for a live or spoof face image.

◇ We achieve the state-of-the-art performance on conventional face anti-spoofing databases.

2. Prior Work

We review papers in three relevant areas: traditional face anti-spoofing methods, CNN-based methods, and image depth estimation.

Traditional face anti-spoofing methods Most prior work utilizes hand-crafted features and adopts shallow learning techniques (e.g., SVM and LDA) to develop an anti-spoofing system. A great number of work pay attention to the texture differences between the live faces and the spoof ones. Common local features that have been used in prior work include LBP [30, 15, 16], HOG [25, 47], DoG [42, 36], SIFT [34] and SURF [9]. However, the aforementioned features to detect texture difference could be very sensitive to different illuminations, camera devices and specific identities. Researchers also seek solutions on different color spaces such as HSV and YCbCr [7, 8], Fourier spectra [28] and Optical Flow Maps (OFM) [5].

Additionally, some approaches attempt to leverage the spontaneous face motions. Eye-blinking is one cue proposed in [32, 41], to detect spoof attacks such as paper attack. In [24], Kollreider et al. use lip motion to monitor the face liveness. Methods proposed in [11, 12] combine audio and visual cues to verify the face liveness.

CNN-based methods CNN have proven to successfully outperform other learning paradigms in many computer vision tasks [26, 27, 22]. In [29, 33], the CNN serves as a feature extractor. Both methods fine-tune their network from a pretrained model (CaffeNet in [33], VGG-face model in [29]), and extract the features to distinguish live vs. spoof. In [46], Yang et al. propose to learn a CNN as a classifier for face anti-spoofing. Registered face images with different spatial scales are stacked as input and live/spoof labeling is assigned as the output. In addition, Feng et al. [17] propose to use multiple cues as the CNN input for live/spoof classification. They select Shearlet-based features to measure the image quality and the OFM of the face area as well as the whole scene area. And in [45], Xu et al. propose an LSTM-CNN architecture to conduct a joint prediction for multiple frames of a video.

However, compared to other face related problems, such as face recognition [27, 43] and face alignment [20], there are still substantially less efforts and exploration on face anti-spoofing using deep learning techniques. Therefore, in this work we aim to further explore the capability of CNN in face anti-spoofing, from the novel perspective of fusing the local texture-based decision and holistic depth maps.

Image depth estimation Estimating depth from a single RGB image is a fundamental problem in computer vision. In recent years there have been a rapid progress of data-driven methods [23], especially deep neural networks trained on large RGB-D datasets [40], as well as weak annotations [10]. Specifically, for face images, face reconstruction from one image [20, 21] or multiple images [37, 38] can also be viewed as one way for depth estimation. However, to the best of our knowledge, no prior work has attempted to estimate the depth for a spoof image, such as print attack. In contrast, our approach estimates depth for both the live face and spoof face, which is particularly challenging since the CNN needs to discern the subtle difference between two cases in order to correctly infer the depth. Note that in this work, we take a more general definition of "depth", which is more of the height map of the face region, rather than the strict object-surface-to-camera distance.

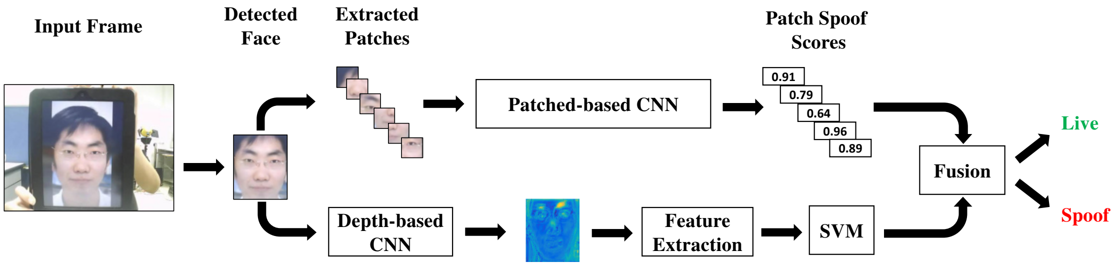

Figure 2: Architecture of the proposed face anti-spoofing approach.

3. Proposed Method

The proposed approach consists of two streams: patch-based CNN, and depth-based CNN. Figure 2 shows a high-level illustration of both streams along with a fusion strategy for combining them. For the patch-based CNN stream, we train a deep neural network end-to-end to learn rich appearance features, which are capable of discriminating between live and spoof face images using patches randomly extracted from face images. For the depth-based CNN stream, we train a fully convolutional network (FCN) to estimate the depth of a face image, by assuming that a print or replay presentation attack have a flat depth map, while live faces contain a normal face depth.

Either the appearance or depth cue can detect face attacks independently. However, fusing both cues has proven to provide promising results. In this paper, we refer to the fusion output as the spoof-score. A face image or video clip is classified as spoof if its spoof-score is above a pre-defined threshold. In the remaining of this section, we explain in detail the two CNN steams used for face anti-spoofing.

3.1. Patch-based CNN

There are multiple motivations to use patches instead of full face in our CNN. First is to increase the number of training samples for CNN learning. Note that for all available anti-spoofing datasets, only a limited number of samples are available for training. E.g., CASIA-FASD only contains 20 training subjects, with 12 videos per subject. Even though hundreds of faces can be extracted from each video, overfitting could be a major issue when learning the CNN due to the high similarities across the frames. Second, when using the full face images as input, traditional CNN needs to resize faces due to varying face image resolutions, where such scaling change might lead to the reduction of the discriminative information. In contrast, using the local patches can maintain the native resolution of the original face images, and thus preserve the discriminative ability. Third, assuming the spoof-specific discriminative information is present spatially in the entire face region, patch-level input can enforce CNN to discover such information, regardless of the patch location. This is a more constrained or challenging learning task compared to using the whole face image.

3.1.1 Input features

CNN is claimed to be a powerful feature learner that is able to map from raw RGB pixel intensities to the discriminative feature representation, guided by the loss function, which is in sharp difference to the conventional hand-crafted features. In our work, one observation is that CNN might also benefit from the hand-crafted features, which are proven to work well for the anti-spoof application. In a way, this is one form of bringing domain knowledge to the CNN learning. This might be especially important for face anti-spoof applications, since without domain knowledge it is more likely for CNN to learn non-generalizable information from the data, rather than the true discriminative feature.

In reviewing hand-crafted features for face anti-spoofing, researchers have been experimenting with several color spaces as input to a feature extraction module to find discriminative descriptors. Typically, the most common color spaces used are RGB, HSV, YC_bC_r, and several combinations among them, such as $HSV + YC_bC_r$ [7]. The RGB has limited applications in face anti-spoofing due to the high correlation between the three color components and the imperfect separation of the luminance and chrominance information. On the other hand, HSV and YC_bC_r are based on the separation of the luminance and the chrominance information, providing additional features for learning the discriminative cues.

In this work, we attempt to use both HSV and YC_bC_r color spaces in the CNN-base methods. Moreover, we also explore several other input feature maps to the CNN including a pixel-wise LBP map, and high-frequency patches. For the pixel-wise LBP map, we use the $LBP_{8,1}$ operator (i.e., $P = 8$ and $R = 1$) to extract the pixel-wise textural features from the face image, and afterwards we randomly extract patches from the texture map. Note that in previous works, LBP is only used to extract histogram descriptors. For the high-frequency patches, the idea is to remove the low-frequency information from the patches which is motivated by the work in [14]. For any given face image $\mathbf{I}$, we subtract the low-pass filtered image of $\mathbf{I}$, which results in a high-frequency image $\mathbf{I}_H = \mathbf{I} - f_{lp}(\mathbf{I})$. An illustration of the various input features explored in our system is in Figure 3. Compared to using RGB alone, providing these input

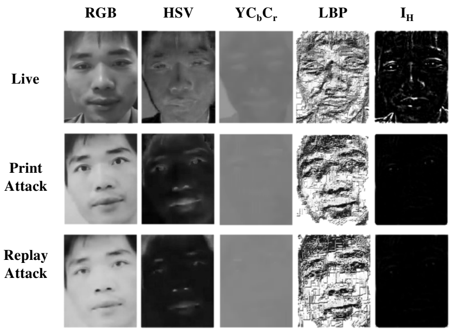

Figure 3: Examples on RGB (G channel), HSV (S channel), YC_bC_r (C_b channel), pixel-wise LBP (LBP of S channel in HSV), high-frequency images (using G in RGB) of both live and spoof face images.

features can facilitate the CNN training.

Based on our experiments, all of the proposed input features are useful representations to learn a CNN capable of distinguishing spoof attacks from live faces. In the experiments section, quantitative results comparing the input features will be presented. For the patch-based CNN, after detecting the face region, we convert the full face image into one of the feature representations, i.e., HSV, and then extract fixed size patches for CNN training and testing.

3.1.2 CNN architecture

A detailed network structure of the patch-based CNN is illustrated in Table 1. Note that a total of five convolutional layers are used followed by three fully connected layers. Following every convolutional layer, we use a batch normalization, ReLU and pooling layers. Softmax loss is utilized in CNN training. Given a training image, we initially detect the face and then crop the face region based on eye positions. After that, several patches are extracted randomly from the face image, such that all patches have the same fixed size. We avoid any rescaling to the original face images for the purpose of maintaining the spoof patterns within the extracted patches. If the face image is a live face, we assign all of its patches a binary label of 1. If the face is a spoof face, the labels of patches are 0.

During testing, we extract patches in the same manner as training. The patch-based CNN will produce spoof scores for every patch in the range of $0-1$. The final result of the image is the average spoof score of all patches. If the presentation attack is in the video format, we compute the average spoof score across all frames.

3.2. Depth-based CNN

In this section, we explain the details of the depth-based CNN. PAs such as printed paper and display, have an obviously different depth compared to the live faces. Therefore, developing a robust depth estimator can benefit the face anti-spoofing.

Table 1: The network structure of patch-based CNN and depth-based CNN. Red texts represent the output of the CNNs. Every convolution layer is cascaded with a ReLU layer. Note that the input size for patch-based CNN is fixed to be 96 × 96. The input size for depth-based CNN is varied from sample to sample. For simplicity, we show the case when the input size is 128 × 128 .

Patch-based CNN			*Depth-based CNN*		
Layer	Filter/Stride	Output Size	Layer	Filter/Stride	Output Size
			Conv-11	$3\times3/1$	$128\times128\times64$
Conv-1	$5\times5/1$	$96\times96\times50$	Conv-12	$3\times3/1$	$128\times128\times64$
BN-1		$96\times96\times50$	Conv-13	$3\times3/1$	$128\times128\times128$
MaxPooling-1	$2\times2/2$	$48\times48\times50$	MaxPooling-1	$2\times2/2$	$64\times64\times128$
			Conv21	$3\times3/1$	$6\times6\times128$
Conv-2	$3\times3/1$	$48\times48\times100$	Conv-22	$3\times3/1$	$64\times64\times256$
BN-2		$48\times48\times100$	Conv-23	$3\times3/1$	$64\times64\times160$
MaxPooling-2	$2\times2/2$	$24\times24\times100$	MaxPooling-2	$2\times2/2$	$32\times32\times160$
Conv-3	$3\times3/1$	$24\times24\times150$			
BN-3		$24\times24\times150$	Conv-31	$3\times3/1$	$32\times32\times128$
MaxPooling-3	$3\times3/2$	$12\times12\times150$	ConvT-32	$6\times6/5$	$37\times37\times128$
Conv-4	$3\times3/1$	$12\times12\times200$			
BN-4		$12\times12\times200$	Conv-41	$3\times3/1$	$37\times37\times128$
MaxPooling-4	$2\times2/2$	$6\times6\times200$	ConvT-42	$6\times6/5$	$42\times42\times128$
Conv-5	$3\times3/1$	$6\times6\times250$			
BN-5		$6\times6\times250$	Conv-51	$3\times3/1$	$42\times42\times160$
MaxPooling-5	$2\times2/2$	$3\times3\times250$	ConvT-52	$6\times6/5$	$47\times47\times160$
FC-1	$3\times3/1$	$1\times1\times1000$			
BN-6		$1\times1\times1000$	Conv-61	$3\times3/1$	$47\times47\times320$
Dropout	0.5	$1\times1\times1000$	ConvT-62	$6\times6/5$	$52\times52\times320$
FC-2	$1\times1/1$	$1\times1\times400$			
BN-7		$1\times1\times400$			
FC-3	$1\times1/1$	$1\times1\times2$	Conv-71	$3\times3/1$	$52\times52\times1$

Based on [14], we believe that high-frequency information of face images is crucial for anti-spoofing, and resizing images may lead to a loss of high-frequency information. Therefore, to be able to handle face images with different sizes, we proposed to maintain the original image size in training the CNN for depth estimation. That is, we train a fully convolutional network (FCN) whose parameters are independent to the size of input face images. The input is face images and the output is the corresponding depth maps. For the live faces, the depth information is from the 3D face shapes estimated using a state-of-the-art 3D face model fitting algorithm [20, 21, 19]. For the spoof faces, the depth information is the flat plain, as assumed by the attack medium's geometry, e.g., screen, paper.

3.2.1 Generating the depth labels

We represent the live face with the dense 3D shape $\mathbf{A}$ as $\begin{pmatrix} x_1 & x_2 & \cdots & x_Q \\ y_1 & y_2 & \cdots & y_Q \\ z_1 & z_2 & \cdots & z_Q \end{pmatrix}$ where z denotes the depth information of the face, and Q is the number of 3D vertices.

Given the face image, the 3D face model fitting algorithm [21] can estimate the shape parameters $\mathbf{p} \in \mathbb{R}^{1\times228}$ and projection matrix $\mathbf{m} \in \mathbb{R}^{3\times4}$. We then use 3DMM

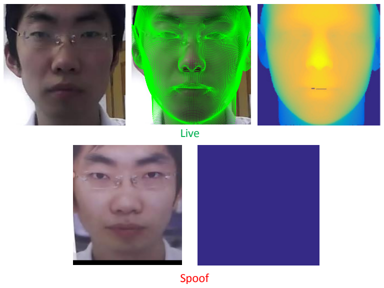

Figure 4: Depth labels for depth-based CNN learning. A live face image, a fitted face model, and the depth label (top row). A spoof face image and the flat plain depth (bottom row).

model [6] to compute the dense 3D face shape $\mathbf{A}$ by

$$\mathbf{A} = \mathbf{m} \cdot \begin{bmatrix} \bar{\mathbf{S}} + \sum_{i=1}^{228} p^i \mathbf{S}^i \\ \mathbf{1}^\intercal \end{bmatrix}, \quad (1)$$

where $\bar{\mathbf{S}}$ is the mean shape of the face and $\mathbf{S}^i$ are the PCA shape bases representing identification variations, e.g., tall/short, light/heavy, and expression variations, e.g., mouth-opening, smile.

After we compute the 3D dense shape of the face, the depth map composes of the z-value for Q vertices from the shape $\mathbf{A}$. In order to obtain a smooth and consistent depth map from discrete z-values from Q vertices, the z-buffering algorithm [31] is applied and the "texture" of the objects is imported as the depth information (i.e. z values). Note that input faces with different sizes would lead to a different range of z values. Hence the depth map $\mathbf{M}$ needs to be normalized before using it as the label for CNN training. In our case, we use the max-min method for normalization.

Examples of depth maps are shown in Figure 4. For spoof faces as well as the background area in the live faces, the z value is equal to 0. Note that for some print attacks, it is possible that the papers are bent. Since it is hard to estimate the actual amount of bending, we also treat the ground truth depth of bent papers as a flat plain.

3.2.2 FCN structure

We employ a FCN to learn the non-linear mapping function $f(\mathbf{I}; \Theta)$ from an input image $\mathbf{I}$ to the corresponding depth map $\mathbf{M}$, where Θ is the network parameter. Following the setting in Sec. 3.1, we use $HSV + YC_bC_r$ features as the CNN input. The depth label $\mathbf{M}$ is obtained in the approach described in the previous subsection. Our FCN network has a bottleneck structure, which contains two parts, downsampling part and upsampling part, as shown in Table 1. The downsampling part contains six convolutional layers and two max pooling layers. The upsampling part consists of five convolutional layers with four transpose convolutional layers in-between for the upsampling purpose. This architecture composes of only convolutional layers without fully connected layer, and each layer is followed by one leaky-ReLU layer. We define the loss function as the pixel-level Euclidean loss,

$$\arg\min_{\Theta} J = \| f(\mathbf{I}; \Theta) - \mathbf{M} \|_F^2 . \quad (2)$$

3.2.3 Depth Map For Classification

The proposed FCN can estimate a depth map for a face image. Since the depth maps used to supervise the training can distinguish between live and spoof images, the estimated depth maps should also have the capability to classify live vs. spoof. To leverage this capability, we train SVM classifiers using the estimated depth maps of the training data.

Specifically, to ensure that the input dimension of SVM is of the same size, the depth map $\mathbf{M}$ is overlaid with a fixed $N \times N$ grid of cells. We compute a mean depth of each local cell and generate a N^2-dim vector, which is fed to the SVM with RBF kernel. Given that resizing the depth map might lose information, we propose to train multiple SVMs with different sizes of N. To properly determine the number of SVMs, we adopt a Gaussian mixture model to fit the distribution of input image sizes. During the testing stage, we feed the testing sample to the SVM, whose input size N is closest to the sample.

Moreover, we can leverage the temporal information given a face video input. For live videos, the depth changes little over time, while the depth of spoof ones can change substantially due to noisy estimation and involuntary hand movement while holding spoof mediums. Hence for a video, we first compute a N^2-dim vector for each frame, and then compute standard deviation of the estimated depth maps of the video. The final feature of a frame feeding to SVM is a $2N^2$-dim vector. Given the SVM output of all frames, we use their average as the final score of the video.

4. Experiments

4.1. Database

We evaluate our proposed method on two PAs: print and replay attacks, using three benchmark databases: CASIA-MFSD [48], MSU-USSA [34], and Replay-Attack [13].

CASIA-MFSD: This database contains 50 subjects, and 12 videos for each subject under 3 different image resolutions and varied lightings. Each subject includes 3 different spoof attacks: replay, warp print, and cut print attacks. Due to the diversity of the spoof types, many previous work [32, 41] that leverage the motion cues such as eye-blinking or shape deformation would fail on this dataset. This dataset partitions the subject space and use 20 subjects for training and 30 subjects for testing.

MSU-USSA: As one of the largest public face spoofing database, MSU-USSA contains $1,000$ in-the-wild live subject images from the Weakly Labeled Face Database [44], and create 8 types of spoof attacks from different devices such as smartphones, personal computers, tablets and printed papers. This dataset covers images under different illuminations, image qualities and subject diversity.

Reaply-Attack: This database contains $1,300$ live and spoof videos from 50 subject. These videos are divided to training, development and testing sets with 15, 15 and 20 subjects respectively. The videos contain two illumination conditions: controlled and adverse. Given the print and replay attacks in this set, the database also divides the attacks into two more types based on whether they use a support to hold the spoof medium, or if the attack is held by a person.

4.2. Experimental Parameters and Setup

Our experiments follow the protocol associated with each of the three databases. For each database, we use the training set to learn the CNN models and the testing set for evaluation in terms of EER and HTER. The Replay-Attack database provides a development set which is only used as a validation set during training to ensure convergence of the network. We select $5,000$ random patches from the development set to validate the training process. For fair comparison on MSU-USSA, we follow the testing protocol in [34], using a subject-exclusive five-fold cross validation, where the subjects are randomly split into five folds.

For the patch-based CNN, we use Caffe toolbox [18], with the learning rate of 0.001, decay rate of 0.0001, momentum of 0.99, and batch size of 100. Before fed into the CNN, the data is normalized by subtracting the mean of training data. Since CASIA and Replay-Attack are video datasets, we only extract two random patches per frame for training. For the images in MSU-USSA, we extract 64 patches from each live face region, and eight patches from each spoof face region. For CASIA and MSU-USSA, a fixed patch size of 96×96 is used. For Replay-Attack, given its low image resolution, the patch size is 24×24. To accommodate the difference in patch sizes, we remove the first two pooling layers for the patch-based CNN.

For the depth-based CNN method, we use Tensorflow toolbox [1], with the learning rate of 0.01 and batch size of 32. The patches are also normalized by subtracting the mean face of training data. When generating the depth labels, we normalize the depth to the range of $0-1$. We use the weighted average of two streams' scores as the final score of our proposed method, where the weights are experimentally determined to be 1 and 0.4 for patch and depth-based streams, respectively.

4.3. Ablation Study

4.3.1 Patch-based CNN analysis

In our work, we explore several input feature maps to train the patch-based CNN, which include different combinations of color spaces, a pixel-wise LBP map, and high-frequency patches. For all of the experiments, we first detect and then crop a face for a given frame. After that we convert the face image into a new feature map as seen in Figure 3, which will then be used to extract patches. Table 2 presents the results on CASIA-FASD when using different combinations of input feature maps. Based on our experiments, we only show the best four combinations of features in this table. From these results, we can clearly see that the $HSV + YC_bC_r$ features has a significant improvement in performance compared to the other features with an EER of 4.44% and an HTER of 3.78%. Moreover, when adding an LBP map to the $HSV + YC_bC_r$ has a negative impact to the CNN learning, which reduces the performance of using $HSV + YC_bC_r$ only by 2.31% HTER. Similarly, when training the patch-based CNN with high-frequency data in the $HSV + YC_bC_r$ images, it also reduces the performance by 1.79% HTER. This shows that the low-frequencies may also provide discriminative information to anti-spoofing.

Table 2: EER (%) and HTER (%) of CASIA-FASD, when feeding different features to patch-based CNN.

Feature	EER (%)	HTER (%)
YC_bC_r	4.82	3.95
$YC_bC_r + HSV$	**4.44**	**3.78**
$YC_bC_r + HSV + LBP$	7.72	6.09
$(YC_bC_r + HSV)_H$	9.58	5.57

4.3.2 Depth-based CNN analysis

The depth map results on the CASIA-FASD testing set are shown in Figure 5. We can find that there is a clear distinction between the depth maps of the live images and those of the spoof images. Compared to the depth label shown in Figure 4, the depth prediction of the live images is still not perfect. However, CNN is attempting to predict the face-like depth, i.e., higher values in the depth map, while the predicted depth of the spoof images to be flat, i.e., lower values in the depth map. In comparison, for the spoof image, there might be certain areas that suffer more degradation and noise from the spoof attack. As we can see from Figure 5, our CNN is still trying to predict some areas with high values in the depth map. However, overall depth patterns of spoof samples are far from those of live samples so that the SVM can learn their difference. Hence, training a CNN for depth estimation is beneficial to face anti-spoofing. Figure 6 shows the mean and standard deviation of the estimated depth maps for all live faces and spoof faces in Reaply-Attack. The differences of live vs. spoof in both mean and standard deviation demonstrate the discriminative ability of depth maps, and supports the motivation of feature extraction for SVM in Sec. 3.2.3.

4.3.3 Fusion analysis

We extensively analyze the performance of our patch-based and depth-based CNNs on CASIA-FASD and report frame-based ROC curves as seen in Figure 7. As mentioned ear-

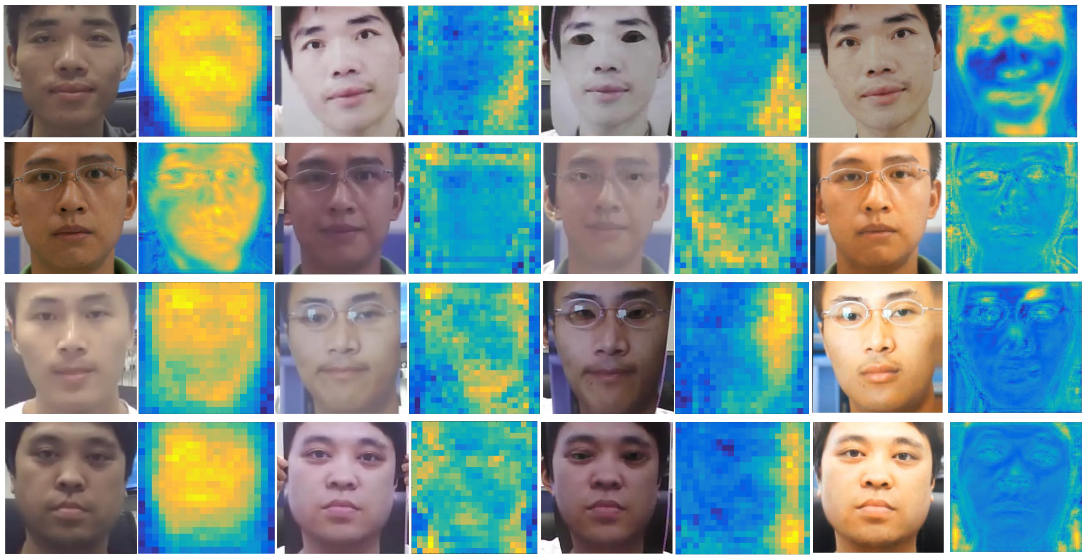

Figure 5: The depth estimation on CASIA-FASD testing subjects. The first two columns are the live images and their corresponding depth maps, the rest six columns are three different types of spoof attacks (print, cut print and video attacks) and their corresponding depth maps.

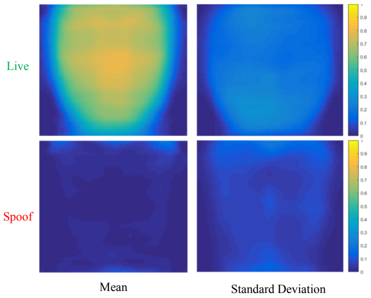

Figure 6: The mean and standard deviation of the estimated depth maps of live and spoof faces, for all testing samples in Reaply-Attack. Note the clear differences in both the mean and standard deviation between the two classes.

lier, CASIA-FASD has three different video qualities and three different presentation attacks, which we use to highlight the differences of our proposed CNN streams.
As a general observation in Figure 7, the fusion-based method tends to perform better than the two streams independently. Similarly, the depth-based methods shows a higher accuracy over the patch-based method across all different qualities and attack types. A clear advantage of using fusion-based method can be illustrated in the normal quality images, where both patch and depth-based methods show an improvement percentage of 1.25 and 0.22 EER.

When analyzing the three different presentation attacks in CASIA-FASD with our proposed methods, the ROC curves are highly correlated. This is because the ROC curve of each attack is an average of the three different video quality performances. The difference in error among the three attacks in the fusion-based results are very small, which illustrate the effectiveness of the proposed method.

4.4. Experimental Comparison

We compare the proposed method with the state-of-the-art methods on CASIA-FASD. Table 3 shows the EER and HTER of ten face anti-spoof methods. Among the different CNN-based methods in Table 3, the temporal features are utilized in a Long Short-Term Memory (LSTM) CNN [45], the holistic features are extracted for classification in [46] and, CNN is used for the feature extraction in [29] and after applying PCA to the response of the last layer, SVM is utilized for classification. The remaining methods in Table 3 are some state-of-the-art methods which utilize hand-crafted features along with a classifier [39, 9, 7]. According to the Table 3, our method outperforms others in EER. This shows the combination of local and holistic features contain more discriminative information. Note that even though patch-based CNN alone has larger errors, its fusion with depth-based CNN still improves the overall performance.

We also test our method on the MSU-USSA database. Not many papers report results in this database because it is relatively new. Table 4 compares our results with [34] which analyzes the distortions in spoof images, and provides a concatenated representation of LBP and color moment. In comparison to [34], our patch-based CNN already achieves 89% reduction of EER. The complementariness of depth-based CNN further reduce both the EER and HTER.

On the Replay-Attack database [13], we compare the proposed method with state-of-the-art methods in Table 5. Although our EER is similar to the prior methods, but the HTER of our method is much smaller (we improve the best HTER by 67.3%), which means we have fewer false acceptance and rejection.

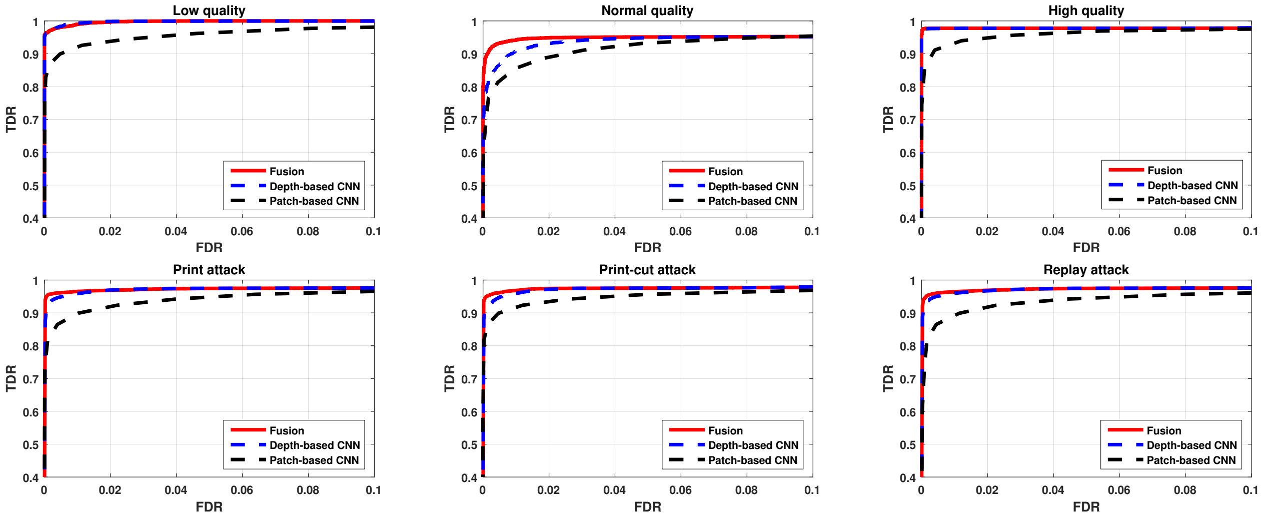

Figure 7: Frame-based ROC curves on CASIA-FASD comparing the fusion method with the patch-based and depth-based CNNs.

Table 3: EER (%) and HTER (%) on CASIA-FASD dataset. The symbol † refers to CNN-based methods.

Method	EER (%)	HTER (%)
Fine-tuned VGG-Face [29]†	5.20	-
DPCNN [29]†	4.50	-
Yang *et al.* [46]†	4.92	-
LSTM-CNN [45]†	5.17	5.93
Boulkenafet *et al.*[7]	6.20	-
Siddiqui *et al.* [39]	3.14	-
Boulkenafet *et al.* [9]	2.80	-
Haralick features [2]	-	1.10
Moire pattern [35]	-	**0**
Our patch-based CNN	4.44	3.78
Our depth-based CNN	2.85	2.52
Our fusion	**2.67**	2.27

Table 4: EER (%) and HTER (%) on MSU-USSA.

Method	EER (%)	HTER (%)
Patel *et al.* [34]	3.84	-
Our patch-based CNN	0.55 ± 0.26	0.41 ± 0.32
Our depth-based CNN	2.62 ± 0.73	2.22 ± 0.66
Our fusion	$\mathbf{0.35} \pm 0.19$	$\mathbf{0.21} \pm 0.21$

5. Conclusions

This paper introduces a novel face anti-spoofing method based on fusing two CNN streams. Unlike the most prior methods in face anti-spoofing that use the full face to detect presentation attacks, we leverage both the full face image and patches extracted from the same face to distinguish the spoof from live faces. The first CNN stream is based on patch appearance extracted from face regions. This stream demonstrates its robustness across all presentation attacks.

Table 5: EER (%) and HTER (%) on Replay-Attack dataset. The symbol † refers to CNN-based methods.

Method	EER (%)	HTER (%)
Fine-tuned VGG-Face [29]†	8.40	4.30
DPCNN [29]†	2.90	6.10
Yang *et al.* [46]†	2.14	-
Boulkenafet *et al.* [7]	0.40	2.90
Boulkenafet *et al.* [9]	**0.10**	2.20
Moire pattern [35]	-	3.30
Our patch-based CNN	2.50	1.25
Our depth-based CNN	0.86	0.75
Our fusion	0.79	**0.72**

The second CNN stream is based on face depth estimation using the full face image. The experiments of this CNN show that our depth estimation can achieve promising results on all image qualities and attacks. Moreover, fusing both CNN streams result in an overall improvement that is compared favorably to the state-of-the-art.

6. Acknowledgment

This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. 2014-14071600012. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 6

[2] A. Agarwal, R. Singh, and M. Vatsa. Face anti-spoofing using haralick features. In *Biometrics Theory, Applications and Systems (BTAS), 2016 IEEE 8th International Conference on*, pages 1–6. IEEE, 2016. 8

[3] A. Anjos and S. Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. pages 1–7, 2011. 1

[4] Y. Atoum, L. Chen, A. X. Liu, S. D. H. Hsu, and X. Liu. Automated online exam proctoring. *IEEE Transactions on Multimedia*, 19(7):1609–1624, July 2017. 1

[5] W. Bao, H. Li, N. Li, and W. Jiang. A liveness detection method for face recognition based on optical flow field. In *Image Analysis and Signal Processing, 2009. IASP 2009. International Conference on*, pages 233–236. IEEE, 2009. 2

[6] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1063–1074, 2003. 5

[7] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face anti-spoofing based on color texture analysis. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 2636–2640. IEEE, 2015. 2, 3, 7, 8

[8] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face spoofing detection using colour texture analysis. *IEEE Transactions on Information Forensics and Security*, 11(8):1818–1830, 2016. 2

[9] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face anti-spoofing using speeded-up robust features and fisher vector encoding. *IEEE Signal Processing Letters*, 24(2):141–145, 2017. 2, 7, 8

[10] W. Chen, Z. Fu, D. Yang, and J. Deng. Single-image depth perception in the wild. In *Advances in Neural Information Processing Systems*, pages 730–738. 2016. 2

[11] G. Chetty. Biometric liveness checking using multimodal fuzzy fusion. In *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*, pages 1–8. IEEE, 2010. 2

[12] G. Chetty and M. Wagner. Audio-visual multimodal fusion for biometric person authentication and liveness verification. In *Proceedings of the 2005 NICTA-HCSNet Multimodal User Interaction Workshop-Volume 57*, pages 17–24. Australian Computer Society, Inc., 2006. 2

[13] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Biometrics Special Interest Group (BIOSIG), 2012 BIOSIG-Proceedings of the International Conference of the*, pages 1–7. IEEE, 2012. 5, 7

[14] A. da Silva Pinto, H. Pedrini, W. Schwartz, and A. Rocha. Video-based face spoofing detection through visual rhythm analysis. In *Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on*, pages 221–228. IEEE, 2012. 3, 4

[15] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel. Lbp- top based countermeasure against face spoofing attacks. In *Asian Conference on Computer Vision*, pages 121–132. Springer, 2012. 2

[16] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *Biometrics (ICB), 2013 International Conference on*, pages 1–8. IEEE, 2013. 2

[17] L. Feng, L.-M. Po, Y. Li, X. Xu, F. Yuan, T. C.-H. Cheung, and K.-W. Cheung. Integration of image quality and motion cues for face anti-spoofing: A neural network approach. *Journal of Visual Communication and Image Representation*, 38:451–460, 2016. 2

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. 6

[19] A. Jourabloo and X. Liu. Pose-invariant 3d face alignment. In *Proc. International Conference on Computer Vision*, Santiago, Chile, December 2015. 4

[20] A. Jourabloo and X. Liu. Large-pose face alignment via cnn-based dense 3d model fitting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4188–4196, 2016. 2, 4

[21] A. Jourabloo and X. Liu. Pose-invariant face alignment via cnn-based dense 3d model fitting. *International Journal of Computer Vision*, pages 1–17, April 2017. 2, 4

[22] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014. 1, 2

[23] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2144–2158, 2014. 2

[24] K. Kollreider, H. Fronthaler, M. I. Faraj, and J. Bigun. Real-time face detection and motion analysis with application in “liveness” assessment. *IEEE Transactions on Information Forensics and Security*, 2(3):548–558, 2007. 2

[25] J. Komulainen, A. Hadid, and M. Pietikainen. Context based face anti-spoofing. In *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, pages 1–8. IEEE, 2013. 2

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1, 2

[27] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997. 1, 2

[28] J. Li, Y. Wang, T. Tan, and A. K. Jain. Live face detection based on the analysis of fourier spectra. In *Defense and Security*, pages 296–303. International Society for Optics and Photonics, 2004. 2

[29] L. Li, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, and A. Hadid. An original face anti-spoofing approach using partial convolutional neural network. In *Image Processing Theory Tools*

and Applications (IPTA), 2016 6th International Conference on, pages 1–6. IEEE, 2016. 2, 7, 8

[30] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Biometrics (IJCB), 2011 international joint conference on*, pages 1–7. IEEE, 2011. 2

[31] T. Matsumoto. Graphics system shadow generation using a depth buffer, Aug. 27 1991. US Patent 5,043,922. 5

[32] G. Pan, L. Sun, Z. Wu, and S. Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007. 2, 5

[33] K. Patel, H. Han, and A. K. Jain. Cross-database face anti-spoofing with robust feature representation. In *Chinese Conference on Biometric Recognition*, pages 611–619. Springer, 2016. 2

[34] K. Patel, H. Han, and A. K. Jain. Secure face unlock: Spoof detection on smartphones. *IEEE Transactions on Information Forensics and Security*, 11(10):2268–2283, 2016. 2, 5, 6, 7, 8

[35] K. Patel, H. Han, A. K. Jain, and G. Ott. Live face video vs. spoof face video: Use of moiré patterns to detect replay video attacks. In *Biometrics (ICB), 2015 International Conference on*, pages 98–105. IEEE, 2015. 8

[36] B. Peixoto, C. Michelassi, and A. Rocha. Face liveness detection under bad illumination conditions. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 3557–3560. IEEE, 2011. 2

[37] J. Roth, Y. Tong, and X. Liu. Adaptive 3d face reconstruction from unconstrained photo collections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4197–4206, 2016. 2

[38] J. Roth, Y. Tong, and X. Liu. Adaptive 3d face reconstruction from unconstrained photo collections. December 2016. 2

[39] T. A. Siddiqui, S. Bharadwaj, T. I. Dhamecha, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha. Face anti-spoofing with multifeature videolet aggregation. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 1035–1040. IEEE, 2016. 7, 8

[40] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. *Computer Vision–ECCV 2012*, pages 746–760, 2012. 2

[41] L. Sun, G. Pan, Z. Wu, and S. Lao. Blinking-based live face detection using conditional random fields. *Advances in Biometrics*, pages 252–260, 2007. 2, 5

[42] X. Tan, Y. Li, J. Liu, and L. Jiang. Face liveness detection from a single image with sparse low rank bilinear discriminative model. *Computer Vision–ECCV 2010*, pages 504–517, 2010. 2

[43] L. Tran, X. Yin, and X. Liu. Disentangled representation learning gan for pose-invariant face recognition. In *In Proceeding of IEEE Computer Vision and Pattern Recognition*, Honolulu, HI, July 2017. 2

[44] D. Wang, S. Hoi, and J. Zhu. Wlfdb: Weakly labeled face databases. *Technical Report*, 2014. 5

[45] Z. Xu, S. Li, and W. Deng. Learning temporal features using lstm-cnn architecture for face anti-spoofing. In *Pattern Recognition (ACPR), 2015 3rd IAPR Asian Conference on*, pages 141–145. IEEE, 2015. 2, 7, 8

[46] J. Yang, Z. Lei, and S. Z. Li. Learn convolutional neural network for face anti-spoofing. *CoRR*, abs/1408.5601, 2014. 2, 7, 8

[47] J. Yang, Z. Lei, S. Liao, and S. Z. Li. Face liveness detection with component dependent descriptor. In *Biometrics (ICB), 2013 International Conference on*, pages 1–6. IEEE, 2013. 2

[48] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *Biometrics (ICB), 2012 5th IAPR international conference on*, pages 26–31. IEEE, 2012. 5