Face anti-spoofing with joint spoofing medium detection and eye blinking analysis

M.Yu. Nikitin [1,2], *V.S. Konushin* [2], *A.S. Konushin* [1,3]
[1] *M.V. Lomonosov Moscow State University, Moscow, Russia,*
[2] *Video Analysis Techonologies LLC, Moscow, Russia,*
[3] *National Research University Higher School of Economics, Moscow, Russia*

Abstract

Modern biometric systems based on face recognition demonstrate high recognition quality, but they are vulnerable to face presentation attacks, such as photo or replay attack. Existing face anti-spoofing methods are mostly based on texture analysis and due to lack of training data either use hand-crafted features or fine-tuned pretrained deep models. In this paper we present a novel CNN-based approach for face anti-spoofing, based on joint analysis of the presence of a spoofing medium and eye blinking. For training our classifiers we propose the procedure of synthetic data generation which allows us to train powerful deep models from scratch. Experimental analysis on the challenging datasets (CASIA-FASD, NUUA Imposter) shows that our method can obtain state-of-the-art results.

Keywords: face anti-spoofing, synthetic data, video analysis, neural networks, deep learning.

Citation: Nikitin MYu, Konushin VS, Konushin AS. Face anti-spoofing with joint spoofing medium detection and eye blinking analysis. Computer Optics 2019; 43(4): 618-626. DOI: 10.18287/2412-6179-2019-43-4-618-626.

Introduction

The problem of face recognition was widely studied for the last several decades and modern facial recognition algorithms already demonstrate superhuman recognition quality. But for real-case scenarios facial recognition systems should not only identify a face in front of the camera correctly, but also detect and process face spoofing attacks, or presentation attacks.

Indeed, the main purpose of a facial recognition system is the accounting of human faces. In case when someone tries to fool the system with artificial face, the system should detect such behavior, and stop further processing. There are many different types of face presentation attacks including print attack, replay attack, 3D masks, etc. As a result, face-based biometric systems can be very vulnerable to such presentation attacks. And in order to make facial recognition systems more secure, special anti-spoofing techniques should be used to classify whether input facial sample is real or fake.

Depending on the type of features used for representation extraction, existing face anti-spoofing approaches can be divided into two groups: liveness-based methods and texture-based methods. Liveness based methods try to detect signs of life by tracking facial parts movements, such as eye blinking or lip movements. These methods are suitable only for static presentation attacks like printing attack or photo replay attack, but inapplicable in case of dynamic attacks (e.g. video replay). Texture based techniques rely on the observation that real and fake face images contain some unique spatiotemporal properties which can help to discriminate between them. Such approaches potentially could be used for detection of both static and dynamic attacks, but because of weak explicit correlation between pixel intensities and attack presence, extracting robust texture features is quite challenging.

Despite of significant success of using deep learning in computer vision tasks, many existing face anti-spoofing methods are still based on hand-engineered features. Obviously, such methods are not optimal, because the usage of hand-crafted features inevitably means the loss of some information. Nevertheless, there exist CNN based approaches for face anti-spoofing, but in order to avoid overfitting most of them either use very simple models or fine-tune existing image classification models.

The main reason of such situation is a lack of training data, which is the key condition for successful application of deep models. Indeed, existing face anti-spoofing datasets are quite small, and not sufficient for training powerful deep model from scratch. At the same time, because of specificity of face spoofing detection task, it is almost impossible to mine fake face images in the Internet.

To avoid this limitation, we propose a new technique for generation of synthetic fake face images. To generate our synthetic data, we have prepared a set of possible spoofing medium templates (printing paper, different models of mobile phones and tablets) and a set of real face images. To produce synthetic image, we insert face image into spoofing medium and place this medium into some background image. All steps are done stochastically, and it allows us to produce arbitrarily large dataset and use it to train powerful face anti-spoofing deep model from scratch. In addition to spoofing medium detection, we propose to use motion-based information. More precisely, we train deep neural network for human eyes openness classification on images, and use it to analyze the presence of eye blinking on videos. In order to combine outputs from two classifiers in complementary way, special fusion strategy is proposed in this paper. Resulting algorithm achieves state-of-the-art results on detecting printing and replay attacks on CASIA-FASD [1] and NUAA Impostor [2] datasets. An overview of our approach is presented in Fig. 1.

The main contributions of this paper can be summarized as follows:

- Our proposed method simultaneously determines the presence of spoofing medium around the face area, and analyzes liveness of this face.

- We propose a new method for generating synthetic fake face images, which allows to train complex deep models for spoofing medium detection from scratch.
- We achieve state-of-the-art performance on two publicly available face anti-spoofing datasets.

The rest of this paper is organized as follows. Section I presents a review of related works. In Section II we describe our approach in details. Our experimental setup and results are reported in Section III. Conclusions are finally drawn in the last section.

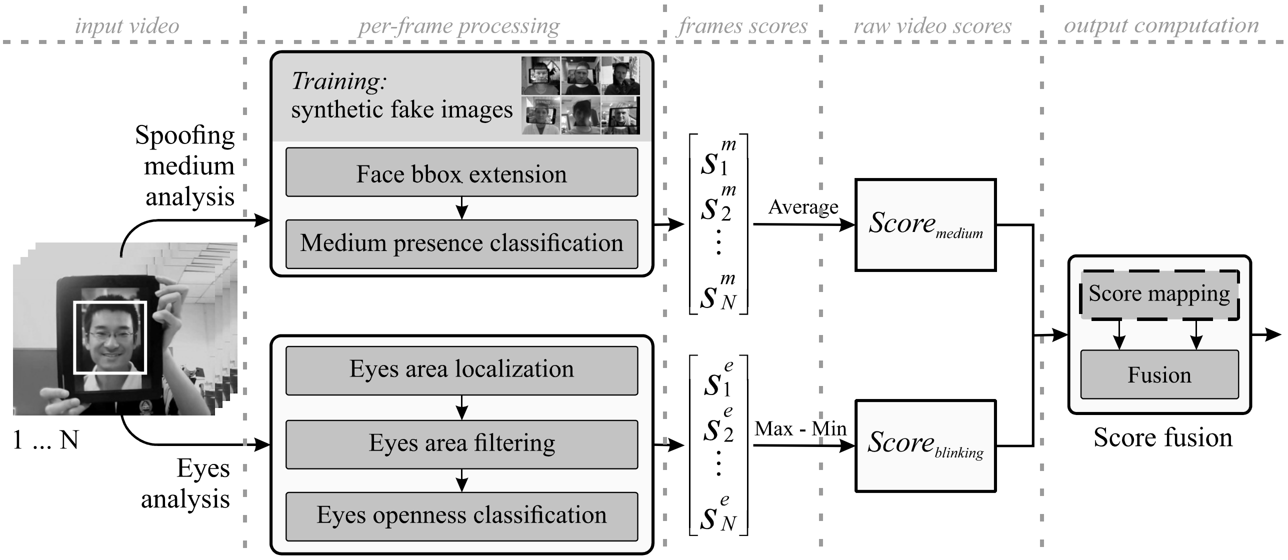

Fig. 1. Overview of our approach. Input video frames are independently processed by two submodules: the first one is trained with synthetic data and performs spoofing medium detection, while the second analyzes eyes area. Per-frame scores from each module are aggregated and resulting values serve as input to the score fusion module, which produces overall liveness score

I. Related work

In this section we review the prior face anti-spoofing approaches in four groups: texture-based methods, liveness methods, existing multi-cues approaches, and applying of Convolutional Neural Networks (CNN) for face anti-spoofing.

A. Texture analysis for face anti-spoofing

Such methods are based on the assumption that real and fake faces images contain different texture patterns. Apparently, this is because of face recapturing process, which leads to facial texture quality degradation and differences in reflectance properties. In order to capture such implicit texture patterns, a great number of prior work utilize hand-engineered local texture features, such as HOG [3], [4], DoG [1], [2], [6], SIFT [7], SURF [12], or LBP variations [8 – 11], [31], and adopt traditional classifiers such as Support Vector Machines or Softmax classifiers. To extract more discriminative features and get over the influence of illumination some papers propose not to use common RGB color space, and switch to another input domain, such as HSV [7], [9], [13] or YCbCr [9], [13].

B. Face liveness detection

The liveness detection approach aims at detecting vital signs through analyzing human face motion. The reason why motion analysis is a good spoofing countermeasure is that the movement characteristics of rigid planar objects, like photo prints or tablet display, differs significantly from that of real human faces which are complex non-rigid 3D objects. The most common temporal cues used for liveness detection are facial expression alterations [8], [16], eye blinking [14 – 16], [38], and lip movements [17], [18]. Instead of analyzing specific face parts, some approaches use more general motion features and techniques, like optical flow analysis [19], [20], [22], Euler motion magnification [21], local speed patterns [34], Shearlet-based features [22] and recurrent neural networks [23]. Motion-based methods are usually effective to printing and photo replay attacks, but become powerless under video replay attacks and printing attacks with cut out eyes or lips area.

C. Multi-cues face anti-spoofing

Some works present multi-cues methods that integrate evidence from multiple complementary sources. The idea behind such approach is that anti-spoofing features relying on a single cue can be not effective for all kinds of presentation attacks, while combining cues of different nature can result in more robust anti-spoofing system. So, in [35] and [22] authors proposed to combine some texture features (color histograms, Gabor features, LBP, etc.) with motion analysis (local face motion, correlation between the face and background regions), and fuse results at a score level. Pereira et al. [5] also analyzed texture and motion cues, but combined results through features concatenation. In [22] Feng et al. built multi-cues face anti-spoofing system, which aggregates image quality features and optical flow maps in one neural network. Authors of [32] proposed to fuse deep features and diffusion-kernel features to achieve better performance. Recently, Atoum et al. [13] proposed to use depth information in addition to texture in their two-stream convolutional neural network model.

D. CNN based face anti-spoofing

Convolutional Neural Networks have proven to be effective in a wide range of computer vision problems. As for facial analysis, modern deep learning approaches consistently outperform other learning paradigms in the tasks of face detection and face recognition. However, in the face anti-spoofing area, the use of deep convolutional networks is still not so common, and has not been sufficiently investigated. Most of the existing CNN-based approaches rely on the assumption of existing discriminative texture patterns for live and spoof examples, and try to automatically compose a model for extraction of such features. In [25], Yang et al. propose to learn AlexNet and use output from one of its hidden layers as input to SVM classifier. The work of [23] adopts a CNN-LSTM architecture for joint analysis of multiple video frames. CNN based methods reliably outperform most of the methods based on hand-engineered features, but are likely prone to overfitting. Because of insufficient amount of available data, learning robust deep models for face anti-spoofing from scratch can be hard and usually result in their weak generalization ability [25], [16]. That is why in some works [16], [26] models are pretrained on ImageNet and then only fine-tuned on dataset of target domain. Authors of [22] and [33] propose to combine hand-crafted features with deep learning techniques to ease the process of model training and avoid overfitting. In this work, we further explore the use of CNNs in face anti-spoofing task, and propose a way to construct robust deep classifiers for analyzing the presence of spoofing medium and human eye blinking.

II. Proposed method

In this paper we propose a new method of detection of 2D facial spoofing attacks which can be presented with face photo print or replay on some device. In order to deal with such attacks, we propose to analyze two types of complementary features: visibility of spoofing medium in front of the camera and presence of eye blinking. Our algorithm consists of three modules: spoofing medium detector, eye blinking detector and scores fusion module. Detailed description of each module is given in the following subsections.

A. Spoofing medium presence detection

The idea behind the medium detection is very simple and natural: if one object appears in front of some other objects, there inevitably will become visible texture discontinuities on the boundaries of the foreground object. It is true both for photo prints and video replays from mobile phones or tablets, and we propose to create an algorithm for detecting the presence of such discontinuities on face images.

To the best of our knowledge, our work is the first attempt to perform explicit spoofing medium detection to determine face presentation attacks, while some existing face anti-spoofing works also use context information. So, in [3] and [25] texture analysis on extended face bounding boxes was performed. Authors of [22] used optical flow maps computed on uncropped frames to capture spoofing medium movements.

In recent years huge progress in computer vision area has been made with the help of deep convolutional networks, and our spoofing medium detector is also based on deep CNN architecture. One of the main reasons why deep learning is so successful is the availability of large task-specific datasets (e.g. ImageNet for object classification), but for face anti-spoofing task existing datasets are relatively small and hardly can be used for training complex deep models. To struggle with this problem, we propose a new method of synthetic data generation, which allows us to generate arbitrarily large amount of fake face images and therefore to train deep face anti-spoofing models with good generalization ability. To our knowledge it is the first attempt to use synthetic data for face spoofing medium detector training.

1) Synthetic data generation: We propose the algorithm for synthetic data generation, which allows to create images with print and replay face presentation attacks. The process of generating synthetic fake face image is the following (see visualization in Fig. 2):

1) Take an arbitrary face image and localize its bounding box using facial detector.
2) Apply some jittering to the found facial area, like random shifting, scaling and rotation. Crop face image using augmented bounding box.
3) Randomly choose a spoofing medium template image and paste the face image from the previous step inside of it. (A set of spoofing medium templates should be prepared in advance. In our case it consists of a blank image of standard size, and normalized images with mobile phones and tablets of different mades).
4) Take an arbitrary background image and choose 4 points on it. These points will define the area where spoofing medium with face image will be inserted, and should be chosen so that the ratio between inter-points distances will roughly match the ratio between medium template image sides. For better realism you can use image with human upper body and choose points for medium insertion in the area of human's head.
5) Estimate homography from point correspondences and insert medium template with face image to the background image using projective transformation.

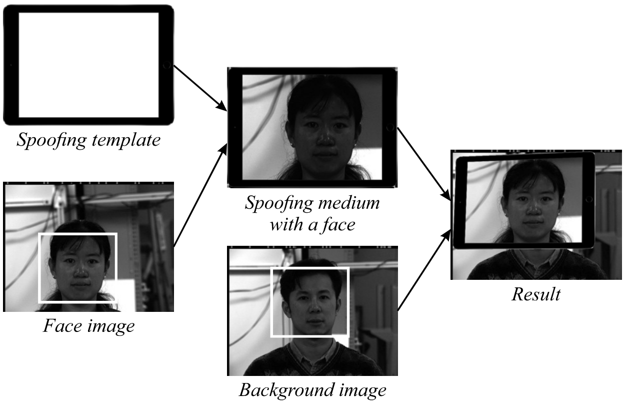

Fig. 2. Synthetic data generation process

Because of indeterministic nature of the process, having large enough dataset for faces sampling and repeating these steps, we can generate any number of synthetic fake face images. Unlimited amount of training data allows us to train complex deep classification models even from scratch.

In this work we consider only attacks when spoofing medium is entirely visible, and our synthetic data generation process is adjusted accordingly. This condition is usually kept in real-life cases, since in practice if one can't see what the camera sees, it is not easy to place spoofing medium in a way that medium's edges become invisible.

2) Deep medium presence classifier: The ability to generate unlimited number of train images allows us to train complex deep classification models without thinking too much about overfitting, which is a big problem for many existing approaches. As a result, relying on the analysis of the spoofing medium presence, we can get more accurate binary face spoofing classifier compared to the previously proposed.

To train our model we prepare the dataset with human face images of two classes, the first one is real human photos and the second one is synthetically generated face spoofing images. We use extended face bounding box area as input to our deep neural network. The reason why we extend the input area is that it allows to model larger variety of face size and position relatively to the spoofing medium. The exact value of bounding box extension has a strong impact on the resulting classification quality. Through the series of experiments, we found that the optimal extension factor is 4.0, and use such value in all subsequent experiments. Resulting classifier can process only single images, and to classify video sequence as real or fake, we process each frame independently and average predictions.

B. Eye blinking detection

The second feature we analyze is the presence of eye blinking on a face video sequence. To determine whether eyes blink on a video we create a simple yet effective eyes openness classifier which is applied for each video frame. This classifier outputs the probability that eyes are open on the input image, and to classify the whole video sequence we analyze the difference between the maximum and the minimum probabilities. If the resulting difference is big enough, it means that there is at least one transition between open and closed eyes in the input video sequence. See the Fig. 3.

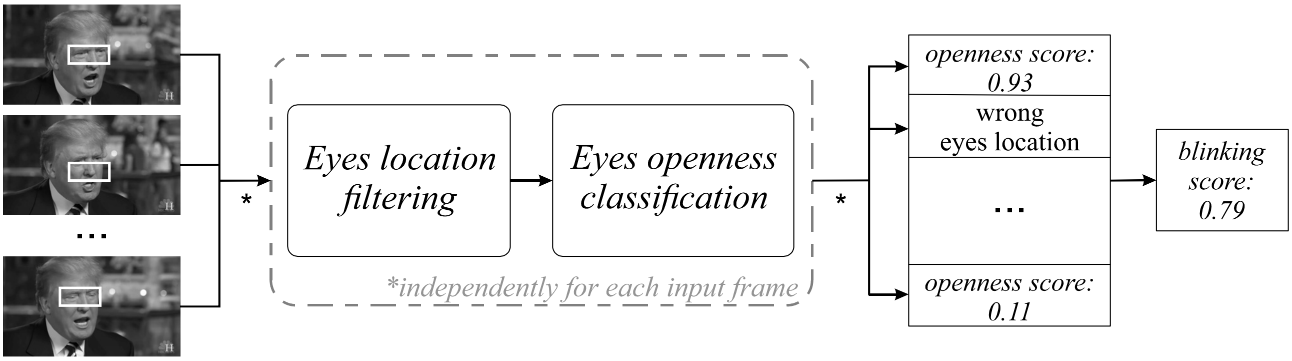

Fig. 3. Eye blinking detection scheme. Eyes area on each frame is detected using facial feature points detector. Detected areas are filtered, and erroneous detections are dropped.
Correct patches are used as input to eyes openness classifier and processed independently. Resulting set of openness scores is aggregated to obtain video blinking score

1) Eyes openness classifier: To train the eyes openness classifier we prepared a small dataset with images of two classes: open eyes and closed eyes. Our dataset was com-posed of publicly available Closed Eyes In The Wild dataset [29] and manually labeled human face images. We use a rectangular patch around the human eyes as input to the openness classifier. In order to precisely localize eyes position, we use facial points detector from the Tevian FaceSDK library [27].

The task of classification of tightly localized eyes as open or closed is relatively simple, but sometimes facial points detector can make mistakes and output points locations can be erroneous. Usage of such wrong points detections for eyes openness classification is impermissible because the output of classifier on the wrong input area is completely unpredictable. That is why we use additional helper network for eyes area detection filtering to increase the stability of eye blinking detection.

2) Facial points filtering: The facial points filtering is conducted every time before eyes openness classification and its main goal is to analyze image patch with eyes area and classify it as wrong or correct. In case the eyes area is wrongly detected, this frame is dropped and isn't used for further analysis.

To train filtering network we automatically generated a dataset with two classes: positive class images with correct eyes area and negative class images with strongly shifted eyes area or even with non-eyes area. To crop images of positive class we use exactly the same cropping strategy as for eyes openness classifier, and to crop images of negative class we use randomly shifted facial points. The range of shifting was chosen experimentally, through analysis of the quality degradation of eyes openness classifier depending on the points shifting distance.

C. Fusion strategy

Both of the discussed above approaches (spoofing medium detection and eye blinking detection) could be used for face spoofing detection independently. But using two different types of features simultaneously could improve the quality and robustness of face anti-spoofing system.

Since our classifiers process each video frame independently, aggregate their results and output scalar liveness scores, then the only possible way of their combination is the fusion at score level. We studied 3 different score fusion approaches: logistic regression, scores multiplication, and scores minimum.

Fusion through logistic regression means that we estimate the significance of each classifier and calculate its contribution to the final result according to its weight. The scores multiplication approach assumes that classifiers are tantamount, and the final score will be high only if all classifiers output high score. The scores minimum approach can be considered as harder version of the scores multiplication fusion method: the final decision is based only on the least confident classifier. All three fusion methods written mathematically are presented in Table 1. Here Score$_{\text{medium}}$ and Score$_{\text{blinking}}$ mean outputs from spoofing medium presence classifier and eye blinking classifier, respectively.

Table 1. Investigated score fusion formulas

Fusion method	Formula for final score computation
Logistic regression	$w_1 * Score_{medium} + w_2 * Score_{blinking}$
Scores multiplication	$\widehat{Score}_{medium} * \widehat{Score}_{blinking}$
Scores minimum	$\min(\widehat{Score}_{medium}, \widehat{Score}_{blinking})$

Because of the fact that the raw outputs of two independent classifiers can't be compared explicitly, we propose to map their output scores to a common scale, before the fusion (this step is necessary for scores multiplication and scores minimum approaches, while can be omitted in case of logistic regression fusion). In Table 1 mapped score values are marked with the hat.

In order to perform such mapping, we manually chose a set of key points in terms of error rates, and for each classifier we estimate a piecewise linear transform using these points. This transformation maps raw classifier score to a [0, 1] interval. The correspondence we used between error rates and mapped score values is presented in Table 2. Here *FNR* stands for *False Negative Rate*, and *FPR* stands for *False Positive Rate*. In our experiments we used additional held-out dataset for scores mapping weights computation.

Table 2. Correspondence used for score mapping estimation

Key point	Mapped score
FNR = 0.0001	0.1
FNR = 0.001	0.2
FNR = 0.01	0.3
FNR = 0.1	0.4
FNR = FPR	0.5
FPR = 0.1	0.6
FPR = 0.01	0.7
FPR = 0.001	0.8
FPR = 0.0001	0.9

III. Experiments

In this section, we first briefly describe two benchmark datasets and the setup that was used in our experiments. Then, experimental results analysis is provided.

A. Datasets

To evaluate the effectiveness of our face anti-spoofing approach, we performed a set of experiments using two publicly available datasets: CASIA Face Anti-Spoofing Database [1] and NUAA Photograph Imposter Database [2]. These benchmark datasets are quite challenging and include recordings of various 2D face spoofing attack attempts. In addition, most fake videos in these datasets contain at least partly visible spoofing medium, while it is not the case for many other existing datasets [36], [37].

The CASIA Face Anti-spoofing database (CASIA-FASD) contains 50 genuine subjects, and for each subject there are 12 real and fake videos captured under varying imaging quality and camera resolution. Three presentation attacks were designed: warped photo attack, cut photo attacks (eye blinking is presented), and video replay attacks. In total, the dataset includes 600 video recordings, which are divided into two subject-disjoint subsets for training and testing (240 and 360, respectively).

The NUAA dataset consists of 12 614 images extracted from real and fake face videos of 15 subjects captured on a webcam in three sessions with different place and illumination conditions. Face spoofing attacks presented in the dataset are based on hand-held printed face photos. All images are divided into training and test non-overlapping sets. Training set includes 1 743 real images and 1 748 fake images, and test set comprises 3 362 real images and 5 761 fake face images.

In our experiments we followed standard protocols of the datasets, and measured Equal Error Rate (EER) and classification accuracy, on CASIA-FASD and NUAA datasets respectively. On CASIA-FASD database, in our experiments we exclude 1 video record from each subject, in order to perform correct evaluation. We call the resulting dataset “fair subset”. Excluded videos contain video replay attacks with tablet display placed very close to camera, such that medium boundaries are invisible. In such conditions none of our classifiers can detect spoofing attempt even theoretically, that is why the quality of our approach drops on the full dataset.

B. Implementation details

In this section we describe in details neural net architectures, training datasets, and optimization parameters we used during our experiments.

1) Net architectures: The neural architectures we used in our experiments are quite simple yet effective. For medium presence detection we use 112×112 color images as input, and the classification network is based on the ResNet architecture [28]. It consists of 10 convolutional and one fully connected layers. Each convolutional layer is followed by batch normalization and ReLU activation layers. The network for eyes openness classification takes 50×100 color image as input and uses plain convolutional architecture with max-pooling layers. Rectified linear activations (without any normalization) are used after each convolutional layer. The same architecture is used for eyes area filtering. At the output of each network there is fully connected layer with 2 neurons for binary classification. The detailed configurations of our networks are presented in Table 3 and Table 4. We use the following notation:

- ”*ConvX* [3×3, 64], S1” denotes convolution layer with 64 filters of size 3×3 and stride = 1.
- ”*PoolX* [2×2, MAX], S2” stands for max-pooling layer with 2×2 pooling area, and stride = 2.

- ”*FC5* 512” is the fully connected layer with 512 hidden units.
- Residual blocks are shown in two-line brackets.

Table 3. Neural architecture: spoofing medium detection

Layer name	Layer parameters
Conv1.1	$[3\times3,48]$, $S2$
Conv2.1	$[3\times3,96]$, $S2$
Conv2.2	$\begin{bmatrix}3\times3,96\\3\times3,96\end{bmatrix}$, $S1$
Conv3.1	$[3\times3,192]$, $S2$
Conv3.2	$\begin{bmatrix}3\times3,192\\3\times3,192\end{bmatrix}$, $S1$
Conv3.3	$\begin{bmatrix}3\times3,192\\3\times3,192\end{bmatrix}$, $S1$
Conv4.1	$[3\times3,384]$, $S2$
FC5	512

Table 4. Neural architecture: eyes openness classification and eyes area filtering

Layer name	Layer parameters
Conv1	$[3\times3,32]$, $S1$
Pool1	$[2\times2,MAX]$, $S2$
Conv2	$[3\times3,64]$, $S1$
Pool2	$[2\times2,MAX]$, $S2$
Conv3	$[3\times3,96]$, $S1$
Pool3	$[2\times2,MAX]$, $S2$
Conv4	$[3\times3,128]$, $S1$
Pool4	$[2\times2,MAX]$, $S2$
Conv5	$[3\times3,256]$, $S1$
Pool5	$[GLOBAL,AVERAGE]$

Considering computation complexity of neural network forward pass, spoofing medium presence classifier has complexity of 500 MFLOPs, while eyes openness classification and face area classification networks have complexity of 65 MFLOPs each.

2) Training datasets: To train our spoofing medium detector we have prepared 45k images of real human faces and generated 135k synthetic fake face images. As a result, we have a dataset of total size of 180k images with 1:3 ratio of real and fake faces. During synthetic images generation we use 3 spoofing medium templates: Apple iPad tablet, Samsung Galaxy smartphone, and an empty template with fixed aspect ratio (to simulate printing photo attacks). Examples of generated fake images are presented in Fig. 4.

For eyes openness classifier training we combined the Closed Eyes In The Wild dataset [29] with our own set of eyes images. We collected 82 short videos with frontal human faces of 11 subjects, extracted its frames with the rate of 10 fps, and manually labeled them depending on whether eyes are opened or closed. Resulting dataset comprises 4 195 open eyes images and 2 098 closed eyes images.

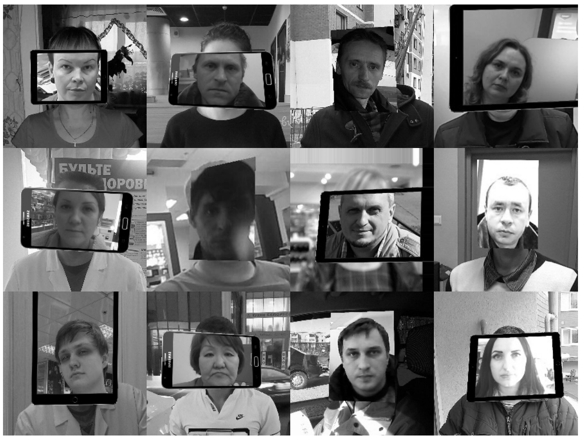

Fig. 4. Examples of generated fake face images

The dataset we used to train eyes area filtering network is based on the eyes openness dataset and was constructed automatically. Namely, for each image from the eyes openness dataset we derived two images: the first one is the original image itself, and the second one is randomly shifted image crop. As a result, training dataset contains both open and closed eyes images in positive and negative class. The shifting distance was chosen experimentally, and to determine lower boundary we studied eyes openness classification accuracy depending on shifting distance. So, for the input image size of 50×100 pixels, we sample shifting distance in the range between 9 and 40 pixels.

In order to increase data variety, we enable image augmentation during training. In all our experiments we use standard set of augmentations which includes random mirroring, random cropping, and random rotation.

3) Training parameters settings: We built our networks using MXNet framework [30] and trained them by optimizing cross-entropy loss. In all our experiments we used stochastic gradient descent with momentum $= 0.9$ and batch size of 512 examples. We train our spoofing medium presence classifier for 40k iterations, starting with learning rate of 0.001 and decreasing it every 10k iterations. To train our eyes openness classifier and face area classifier we initialize learning rate to 0.01 and decrease it every 5k iterations, with the total of 25k iterations. For all our networks learning rate decreasing coefficient was set to 0.1.

We performed all our experiments on a desktop with Intel Core i7-4770 (3.40 GHz) CPU, NVIDIA TITAN X GPU, and 32 Gb RAM. It takes about 4 hours to train spoofing medium presence classifier, and about 30 minutes to train eyes openness classifier or face area classifier.

C. Experimental results

In this section we present and discuss results we have obtained during our experiments with spoofing medium detector by itself, and with its combination with eye blinking detector.

1) Spoofing medium detector evaluation: First of all, we present image-level results for our spoofing medium presence detector. In this experiment we independently classify each video frame as containing real or fake human face. It is a common way of evaluation of face anti-spoofing methods.

Here we compare results of two different models: one was trained only using synthetic fake data, and to get another one we fine-tuned the first model using training subset of the corresponding database. To fine-tune our models we set learning rate to 1e-5 and train them for 5k iterations. Results are presented in Table 5 and Table 6. For fair comparison with existing methods on CASIA-FASD, we evaluated them on the same testing subset which was used for our method (described in subsection III-A), while using the full training set. In this CASIA-FASD experiment we used author's implementations of [9] and [12], and our own implementation of [25].

Table 5. Frame-level results on CASIA-FASD

Method	EER, fair subset	EER, full
Spoofing medium detector, only synthetic data	0.0701	0.1156
Spoofing medium detector, fine-tuned (CASIA-train)	**0.0144**	0.0432
SURF + Softmax [12]	0.0272	0.0280
LBP + SVM [9]	0.0701	0.0620
CNN + SVM [25]	0.0436	0.0492

As can be seen from the tables 5 and 6, our fine-tuned models reach state-of-the-art results. Moreover, by training only on synthetic data we can get competitive performance simultaneously on both testing datasets. Such behavior proves high generalization ability of our approach. To better visualize importance of pretraining with synthetic data, we conduct additional experiment, where the same network architecture was trained from scratch on the CASIA training set. Resulting network demonstrates pretty good results on the CASIA test set (EER = 0.0821), but the classification accuracy on the NUAA dataset drops significantly (64.7 %). So, it proves the fact that using of small training dataset leads to overfitting, and complete inapplicability of the resulting model in real life. At the same time, by using our synthetic data we can pretrain model with good generalization ability, and by additional training on the target data we can just slightly tune it.

Table 6. Frame-level results on NUAA Impostor Database

Method	Accuracy, %
Spoofing medium detector, only synthetic data	98.5
Spoofing medium detector, fine-tuned (NUAA-train)	99.2
SPMT [31]	98.0
ADKMM [32]	**99.3**
ND-CNN [33]	99.0
DS-LSP [34]	98.5
CDD [4]	97.7
DoG-SL [6]	94.5

2) Video-level face anti-spoofing: As we mentioned in previous section, existing face anti-spoofing methods mostly perform classification at frame-level. Obviously, such evaluation is conceptually incorrect, because in practice face recognition systems usually have video sequence as its input instead of a single frame. That is why in this paper we present video-level evaluation results as well.

Table 7. Video-level results: spoofing medium detector

Method	CASIA-FASD, EER	NUAA, Accuracy
Spoofing medium detector, only synthetic data	0.0542	98.8%
Spoofing medium detector, fine-tuned	0.0083	100%

From Table 7 it can be seen that the proposed spoofing medium detector classifier demonstrates good video-level quality. What's more, fine-tuned models reach near perfect results.

3) Combined model evaluation: By incorporating evidence from eye blinking classifier we can further improve video-level classification quality. We performed score fusion strategies comparison on CASIA database. During this set of experiments, we used spoofing medium classifier trained only on synthetic data, in order to better demonstrate how each score aggregation approach influences on resulting quality. Results are presented in Table 8.

Table 8. Score fusion strategies comparison on CASIA-FASD

Method	EER, fair subset	EER, full
Spoofing medium detection only	0.0542	0.1111
Combined: Logistic regression	0.0125	0.0667
Combined: Scores multiplication	**0.0111**	0.0963
Combined: Scores minimum	0.0333	0.1111

According to the results, we can find that for the fair subset no matter what fusion strategy is used, resulting quality becomes better when information from additional classifier is aggregated. Besides, fusion approach based on scores multiplication provides the best results. Moreover, resulting EER value (0.0111) is very close to the EER value obtained by the fine-tuned spoofing medium detector (0.0083).

In addition, we performed evaluation of combined approach using fine-tuned spoofing detector model on the fair subset. For any score fusion approach, resulting method gives EER of 0.0, i.e. it provides perfect separability of real and fake videos. To better visualize how adding of eye blinking information (by multiplication fusion) improves quality even for fine-tuned spoofing medium detector, we plot liveness score distributions for both methods in Fig. 5. It can be seen that combined approach provides more distinct separability between classes.

Conclusion

This paper proposes a novel face anti-spoofing approach based on fusing results of two deep classifiers. The first one performs classification of spoofing medium presence, and in order to avoid overfitting we introduce synthetic data generation procedure. The second classifier analyzes eye blinking and is based on per-frame eyes openness classification. The experiments show that even spoofing medium detector by itself can provide competitive results and high generalization ability for the task of frame-level face anti-spoofing. And by combining two classifiers together we can reach perfect classification quality for video-level scenario on CASIA face anti-spoofing database.

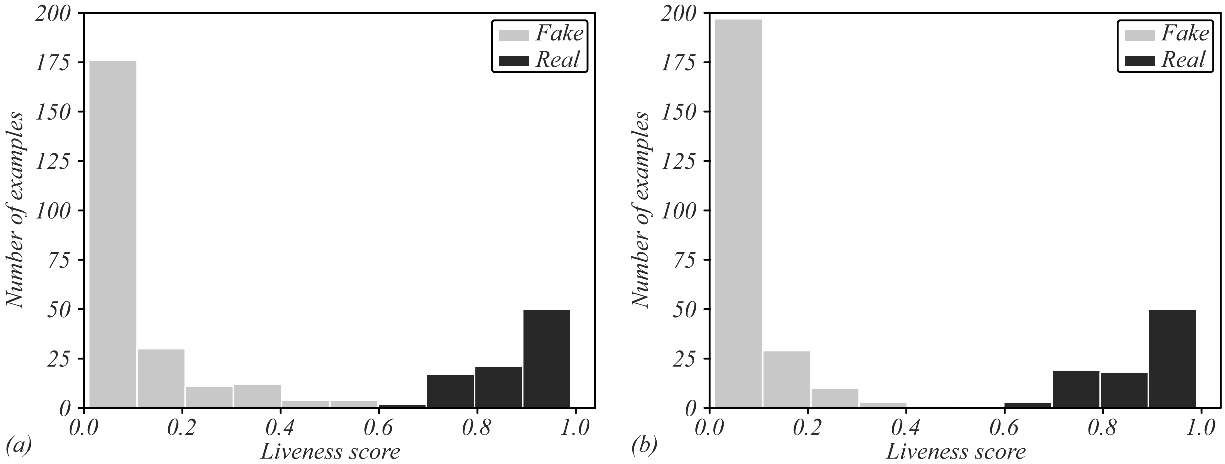

Fig. 5. CASIA-FASD video-level liveness score histograms: (a) spoofing medium detector only; (b) combined method

References

[1] Zhang Z, Yan J, Liu S, Lei Z, Yi D, Li SZ. A face anti-spoofing database with diverse attacks. 5th IEEE International Conference on Biometrics (ICB) 2012: 26-31.

[2] Tan X, Li Y, Liu J, Jiang L. Face liveness detection from a single image with sparse low rank bilinear discriminative model. European Conference on Computer Vision (ECCV) 2010: 504-517.

[3] Komulainen J, Hadid A, Pietikainen M. Context based face anti-spoofing. 6th IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS) 2013: 1-8.

[4] Yang J, Lei Z, Liao S, Li SZ. Face liveness detection with component dependent descriptor. 6th IEEE International Conference on Biometrics (ICB) 2013: 1-6.

[5] De Freitas Pereira T, Komulainen J, Anjos A, De Martino JM, Hadid A, Pietikinen M, Marcel S. Face liveness detection using dynamic texture. EURASIP Journal on Image and Video Processing 2014: 2.

[6] Peixoto B, Michelassi C, Rocha A. Face liveness detection under bad illumination conditions. 18th IEEE International Conference on Image Processing (ICIP) 2011: 3557-3560.

[7] Patel K, Han H, Jain A. Secure face unlock: Spoof detection on smartphones. IEEE transactions on information forensics and security 2016; 11(10): 2268-2283.

[8] Komulainen J, Hadid A, Pietikinen M. Face spoofing detection using dynamic texture. Asian Conference on Computer Vision (ACCV) 2012: 146-157.

[9] Boulkenafet Z, Komulainen J, Hadid A. Face anti-spoofing based on color texture analysis. 22nd IEEE International Conference on Image Processing (ICIP) 2015: 2636-2640.

[10] De Freitas Pereira T, Anjos A, De Martino JM, Marcel S. LBP-TOP based countermeasure against face spoofing attacks. Asian Conference on Computer Vision (ACCV) 2012: 121-132.

[11] De Freitas Pereira T, Anjos A, De Martino JM, Marcel S. Can face anti-spoofing countermeasures work in a real world scenario? 6th IEEE International Conference on Biometrics (ICB) 2013: 1-8.

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

[13] Atoum Y, Liu Y, Jourabloo A, Liu X. Face anti-spoofing using patch and depth-based CNNs. IEEE International Joint Conference on Biometrics (IJCB) 2017: 319-328.

[14] Pan G, Sun L, Wu Z, Lao S. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. 11th IEEE International Conference on Computer Vision (ICCV) 2007: 1-8.

[15] Sun L, Pan G, Wu Z, Lao S. Blinking-based live face detection using conditional random fields. International Conference on Biometrics (ICB) 2007: 252-260.

[16] Patel K, Han H, Jain AK. Cross-database face anti-spoofing with robust feature representation. Chinese Conference on Biometric Recognition 2016: 611-619.

[17] Kollreider K, Fronthaler H, Faraj MI, Bigun J. Real-time face detection and motion analysis with application in liveness assessment. IEEE Transactions on Information Forensics and Security 2007; 2(3): 548-558.

[18] Shao R, Lan X, Yuen PC. Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3D mask face anti-spoofing. IEEE International Joint Conference on Biometrics (IJCB) 2017: 748-755.

[19] Kollreider K, Fronthaler H, Bigun J. Non-intrusive liveness detection by face images. Image and Vision Computing 2009; 27(3): 233-244.

[20] Bao W, Li H, Li N, Jiang W. A liveness detection method for face recognition based on optical flow field. IEEE Image Analysis and Signal Processing (IASP) 2009: 233-236.

[21] Bharadwaj S, Dhamecha TI, Vatsa M, Singh R. Face anti-spoofing via motion magnification and multifeature videolet aggregation. 2014. Source: ⟨https://repository.iiitd.edu.in/jspui/handle/123456789/138⟩.

[22] Feng L, Po LM, Li Y, Xu X, Yuan F, Cheung TCH, Cheung KW. Integration of image quality and motion cues for face anti-spoofing: A neural network approach. Journal of Visual Communication and Image Representation 2016; 38: 451-460.

[23] Xu Z, Li S, Deng W. Learning temporal features using LSTM-CNN architecture for face anti-spoofing. 3rd IEEE Asian Conference on Pattern Recognition (ACPR); 2015: 141-145.

[24] Tronci R, Mutoni D, Fadda G, Pili M, Sirena N, Murgia G, Ristori M, Recerche S, Roli F. Fusion of multiple clues for photo-attack detection in face recognition systems. IEEE International Joint Conference on Biometrics (IJCB); 2011: 1-6.

[25] Yang J, Lei Z, Li SZ. Learn convolutional neural network for face anti-spoofing. arXiv preprint. Source: ⟨https://arxiv.org/abs/1408.5601⟩.

[26] Li L, Feng X, Boulkenafet Z, Xia Z, Li M, Hadid A. An original face anti-spoofing approach using partial convolutional neural network. IEEE Image processing theory tools and applications (IPTA) 2016: 1-6.

[27] Video Analysis Technologies. FaceSDK, facial analysis library. Source: ⟨https://tevian.ru/product/facesdk/⟩.

[28] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016: 770-778.

[29] Song F, Tan X, Liu X, Chen S. Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients. Pattern Recognition 2014; 47(9): 2825-2838.

[30] Chen T, Li M, Li Y, Lin M, Wang N, Wang M, Xiao T, Xu B, Zhang C, Zhang Z. MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems. arXiv preprint. Source: ⟨https://arxiv.org/abs/1512.01274⟩.

[31] Song X, Zhao X, Lin T. Face spoofing detection by fusing binocular depth and spatial pyramid coding micro-texture features. IEEE International Conference on Image Processing (ICIP) 2017: 96-100.

[32] Yu C, Jia Y. Anisotropic diffusion-based kernel matrix model for face liveness detection. arXiv preprint. Source: ⟨https://arxiv.org/abs/1707.02692⟩.

[33] Alotaibi A, Mahmood A. Deep face liveness detection based on nonlinear diffusion using convolution neural network. Signal, Image and Video Processing 2017; 11(4): 713-720.

[34] Kim W, Suh S, Han JJ. Face liveness detection from a single image via diffusion speed model. IEEE Transactions on Image Processing 2015; 24(8): 2456-2465.

[35] Komulainen J, Hadid A, Pietikinen M, Anjos A, Marcel S. Complementary countermeasures for detecting scenic face spoofing attacks. 6th IEEE International Conference on Biometrics (ICB) 2013: 1-7.

[36] Chingovska I, Anjos A, Marcel S. On the effectiveness of local binary patterns in face anti-spoofing. IEEE International Conference of the Biometrics Special Interest Group (BIOSIG) 2012: 1-7.

[37] Boulkenafet Z, Komulainen J, Li L, Feng X, Hadid A. OULU-NPU: A mobile face presentation attack database with real-world variations. 12th IEEE International Conference on Automatic Face Gesture Recognition (FG); 2017: 612-618.

[38] Han YJ, Kim W, Park JS. Efficient eye-blinking detection on smartphones: A hybrid approach based on deep learning. Mobile Information Systems 2018; 2018: 6929762.

Author's information

Mikhail Yurievich Nikitin (b. 1992), post-graduate student, graduated from Lomonosov Moscow State University in 2014, Computational Mathematics and Cybernetics faculty ASVK department, Graphics and Multimedia laboratory. Research interests are computer vision and face recognition. E-mail: *mikhail.nikitin@graphics.cs.msu.ru* .

Vadim Sergeyevich Konushin (b. 1985) graduated from Lomonosov Moscow State University in 2007 and currently works at «Video Analysis Technologies» LLC. E-mail: *vadim@tevian.ru* .

Anton Sergeyevich Konushin (b. 1980) graduated from Lomonosov Moscow State University in 2002. In 2005 successfully defended his PhD thesis in M.V. Keldysh Institute for Applied Mathematics RAS. He is currently associate professor at NRU HSE and Lomonosov Moscow State University. Research interests are computer vision and machine learning. E-mail: *ktosh@graphics.cs.msu.ru* .

Received March 04, 2019. The final version – June 09, 2019.