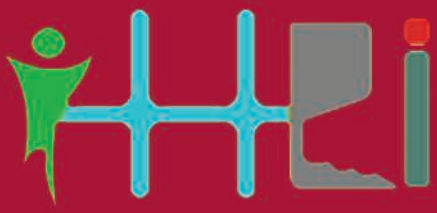
Madhusudan Singh · Dae-Ki Kang ·
Jong-Ha Lee · Uma Shanker Tiwary ·
Dhananjay Singh · Wan-Young Chung (Eds.)

LNCS 12616

Intelligent Human Computer Interaction

12th International Conference, IHCI 2020
Daegu, South Korea, November 24–26, 2020
Proceedings, Part II



Springer

Image Processing and Deep Learning

Adaptive Margin Based Liveness Detection for Face Recognition

Gabit Tolendiyev, Mohammed Abdulhakim Al-Absi, Hyotaek Lim, and Byung-Gook Lee(✉)

Dongseo University, Busan 47011, South Korea
{d0165114,d0185123}@kowon.dongseo.ac.kr
{htlim,lbg}@dongseo.ac.kr

Abstract. Face recognition is currently is becoming the hotspot in the area of deep learning, pattern recognition, and computer vision where it has been broadly utilized in many fields. Facial feature extraction is a key link in the face recognition system. The texture features of human faces are highly discriminative, so extracting the texture features of face images can often get a good classification and recognition effect. Image texture feature extraction methods can generally be classified into four categories: statistical methods, model methods, structural methods, and signal processing methods. Recently, face recognition based person authentication systems have been popular among other biometrics. However, hacking methods are also developed with this methodology. In this paper, we present a margin based liveness detection method (MLDM) for the face recognition system based on texture feature analysis. The fake images captured from a video that has edges generated by differences among different face images of real and fake people images. Moreover, we exploit a convolutional neural network to extract these features and differentiate real and fake face images. Experimental results show that our model has higher accuracy and can efficiently classify real faces and spoofed faces compare with the existing model. The outcome shows that our approach is better than the existing work which is experimentally proven.

Keywords: Face recognition · Face liveness detection · Texture feature · Margin based method · 2D spoofing attack

1 Introduction

Bio-metrics are being broadly used in person authentication systems in the last few decades. However, the most popular bio-metrics are fingerprint identification [7], palm recognition [6], retina recognition [3], IRIS recognition [4], face recognition and etc. are being used at airports, access control systems, payment verification systems and other various fields. Among these bio-metrics face recognition techniques are being used in various fields because of its contactless, fast speed, high accuracy and user-friendliness. Especially in the COVID-19

M. Singh et al. (Eds.): IHCI 2020, LNCS 12616, pp. 267–277, 2021.
https://doi.org/10.1007/978-3-030-68452-5_28

pandemic contactless bio-metric face recognition system is replacing other traditional bio-metrics, such as fingerprint identification and being widely deployed as a person authentication system. However, with the development of this technology, hacking methods are also developing as well. For example, an unauthorized person might use an authorized person's photo to attempt to be authenticated. In computer security it is called a spoofing attack or representation attack. Some examples of the 2D face spoofing attack are illustrated in Fig. 1. For that reason, it is compulsory to develop a new technology that distinguishes a real face image of authorized person and spoofed or fake image is needed.

face spoofing attack examples.PNG

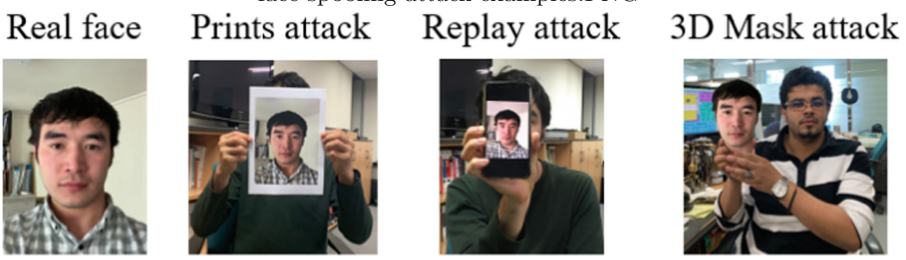


Fig. 1. 2D face spoofing attack examples.

The research work contribution is as follows:

- A database for face liveness detection model is constrained containing real face image and replay spoofing attack image.
- A novel margin based liveness detection method for face recognition is proposed.

The paper organization is as follows. An extensive survey is carried out in Sect. 2. and Sect. 3, proposed a Margin based Liveness Detection Method (MLDM). Section 4 is devoted to the results and experiments. Section 5 is the discussion. Future works is described in Conclusions section.

2 Related Work

2.1 Face Recognition

Face recognition (FR) is more popular than other biometric systems such as fingerprint, palm vein and eye iris recognition. A big reason for using FR is its contactless, non-invasiveness and secureness.

As a face recognition model pretrained OpenFace 0.2.0 is utilized. OpenFace is a free and an open source library for facial recognition with deep convolutional neural networks. It is a Python and PyTorch implementation of facial recognition with deep convolutional neural networks and is based on Computer Vision

and Pattern Recognition Conference 2015 (CVPR) [12]. PyTorch allows the convolutional neural network to be carried in the central processing unit (CPU) or with a parallel computing platform CUDA [1]. Even though OpenFace model is train on the public datasets have orders of magnitude less than private industry datasets, the accuracy of the model is remarkably high on the standard Labeled Faces in the Wild (LFW) public benchmark (Table 1).

Table 1. Accuracy evaluation of OpenFace models on LFW benchmark

Model	Accuracy	AUC
nn4.small2.v1 (Default)	0.9292 ± 0.0134	0.973
nn4.small1.v1	0.9210 ± 0.0160	0.973
nn4.v2	0.9157 ± 0.0152	0.966
nn4.v1	0.7612 ± 0.0189	0.853
FaceNet Paper (Reference)	0.9963 ± 0.009	Not provided

2.2 Liveness Detection

There are several approaches of liveness detection for face recognition, including:

- **Texture analysis techniques**, including computing the Local Binary Patterns (LBPs) over face regions and using an support vector machines (SVMs) to classify the real and fake face images [8].
- **Frequency analysis techniques**, a method of liveness detection by examining the Fourier domain of the face [8].
- **Variable focusing**, a method of liveness detection for 2D fake face images by examining the pixel values variation among two consecutive frames captured in different focuses [9].
- **Heuristic-based algorithms**, a liveness detection method based on blink detection, lip movement and eye movement. These algorithms attempt to track blinks and eye movement to make sure the authenticating person is not holding a printed photo of an authorized person [14].
- **3D face shape**, a method that distinguishes between real faces and printouts, photos, and images of another person by comparing its 3D meshes [10].
- **Combinations of the above methods**, face recognition engineers choose face liveness detection models appropriate to their particular applications.

3 Proposed Approach

In this work, we focused on face liveness detection method against replay spoofing. Because of widely use of smartphone and its availability adversaries possibly

attack with their smartphone rather than using printed face image or 2D mask of the authorized person.

In this paper, the liveness detection model (MLDM) is treated as a binary classification model. It classifies the given face images as real face image or fake spoofed face image. We trained the model on a dataset containing real and spoofed face images (Fig. 2).

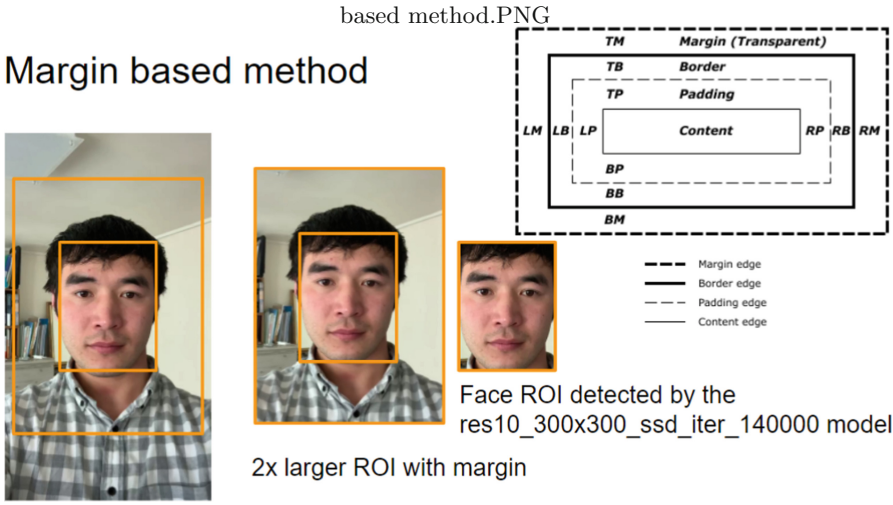


Fig. 2. Margin based method.

Flowchart of the face liveness detection method is depicted in Fig. 3.

Table 2. Model-architecture of MLDM.

LivenessNet64-structural details													
#	Input image			Output			Layer	Stride	Kernel		In	Out	Parameter
1	64	64	3	32	32	16	conv3-16	1	3	3	3	16	448
2	32	32	16	32	32	16	conv16-16	1	3	3	16	16	2320
	32	32	16	16	16	16	maxpool	2	2	2	16	16	0
3	16	16	16	16	16	32	conv16-32	1	3	3	16	32	4640
4	16	16	32	16	16	32	conv32-32	1	3	3	32	32	9248
	16	16	32	8	8	32	maxpool	2	2	2	32	32	0
5	1	1	2048	1	1	64	FC		1	1	2048	64	131136
6	1	1	64	1	1	2	FC		1	1	64	2	130
Total													147,922

Our liveness detection model is extended from [11]. Network architecture of the liveness detection model is presented in Table 2. Input image resized to 64 × 64 and all the pixel intensities are scaled from original range to the range 0 to 1. Data augmentation techniques are also used. A data augmentation object which will generate new face images with random rotation, zoom, width shifting, height shifting, channel shift, shear intensity, horizontal flop and vertical flips. The model is implemented with OpenCV built-in face detection library, Tensorflow and Keras framework. As an optimizer Adam optimization algorithm is used. Training parameters: Learning rate - 1e-4, batch size - 8, number of epochs - 50. Training and testing datasets separated as 75% and 25% respectively. As you can see in Fig. 4, texture features of real and fake images are different. This texture feature difference helps to distinguish real and fake face images (Fig. 5).

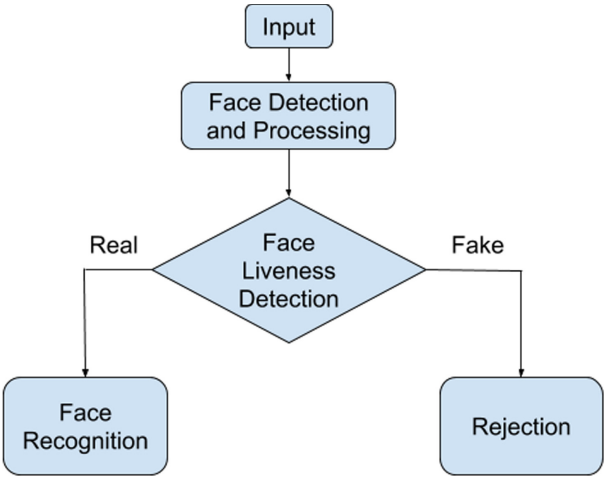


Fig. 3. Flowchart of face liveness detection method (MLDM).

4 Experiments and Results

4.1 Dataset Preparation

The amount of data and its distribution are crucial for training deep learning models. Insufficient data and not well distributed data might affect on generalization ability of the model. As a result, classification accuracy might reduce when model receives as a input a new data. For liveness detection model our dataset has to contains fake and real face images. In order to generate training data for real face images, we recorded a selfie video with the long 20 s of members of our laboratory. Using ResNetSSD face detection model [2] face area detected, cropped with the size of 64 × 64 and stored on the local disk. By skipping every 4 frame consisting face image, 300 face images which is extracted from each video. 20 volunteers are participated in data preparation. Around 300 images

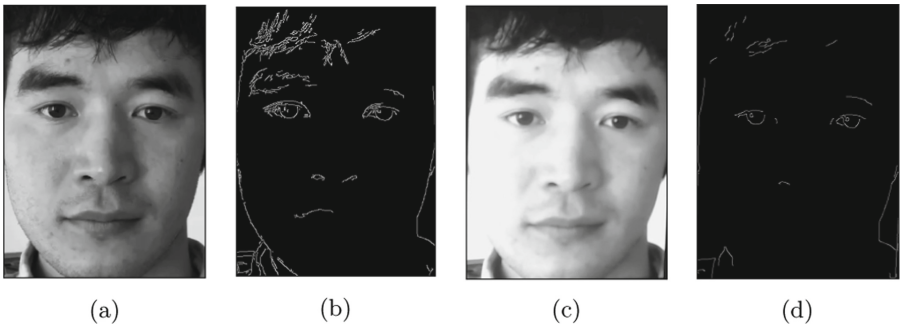


Fig. 4. Samples of input images (**a**) real image and (**b**) its edges, (**c**) fake image and (**d**) its edges.

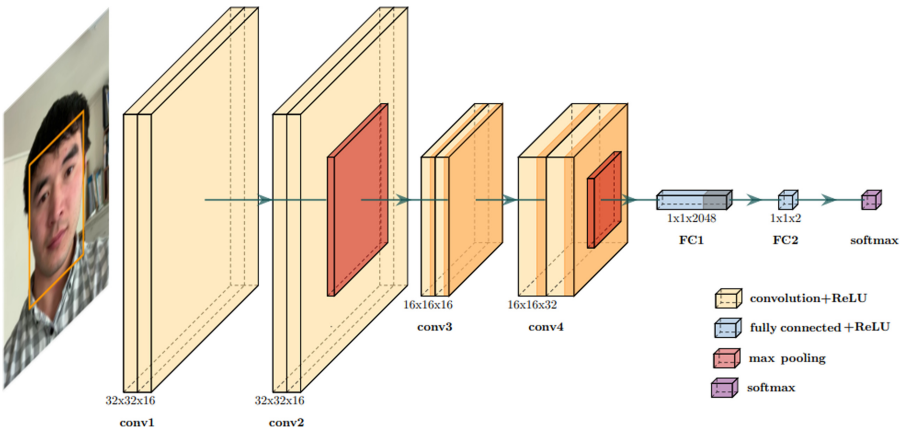


Fig. 5. Model architecture in 3D

Fig. 6. Some of the collected real and fake face images examples for training. The first row shows the real face images and the second row is the fake face images.

were extracted from each persons video. Our dataset contains (total of 15,849 images) 8,328 of real face images and 7,521 of fake face images.

Using the recorded videos, we extracted 3,000 real face images. In order to extract face images, we applied ResNetSSD face detection model [2] described in the preceding section to the whole dataset. Real and fake face images are stored in the separate folder. As a result, 286 real and 3856 fake face images of 20 different people were obtained. The input size of the network was $64 \times 64 \times 3$, therefore all the images are resized to match the input layer of the network. Some samples of the training images are illustrated in Fig. 6.

4.2 Training Model

We trained the our liveness detection model from scratch using the dataset described in the previous section. The dataset was split into a testing set and training set within a proportion of 75% and 25% respectively. To train the liveness detection network, we applied the Adam optimization algorithm with Binary Cross-Entropy Loss function, with the starting learning rate of 0.0004. Training was done on NVIDIA GeForce RTX 2070 16 GB GPU with a batch size of 8. We trained the network for 50 epochs. In prepossessing step, image pixels intensity values are scaled to the range from 0 to 1. Real and Fake label names are strings, they are transformed to integers and the on-hot encode function applied. Also, before training the network, data augmentation operation applied

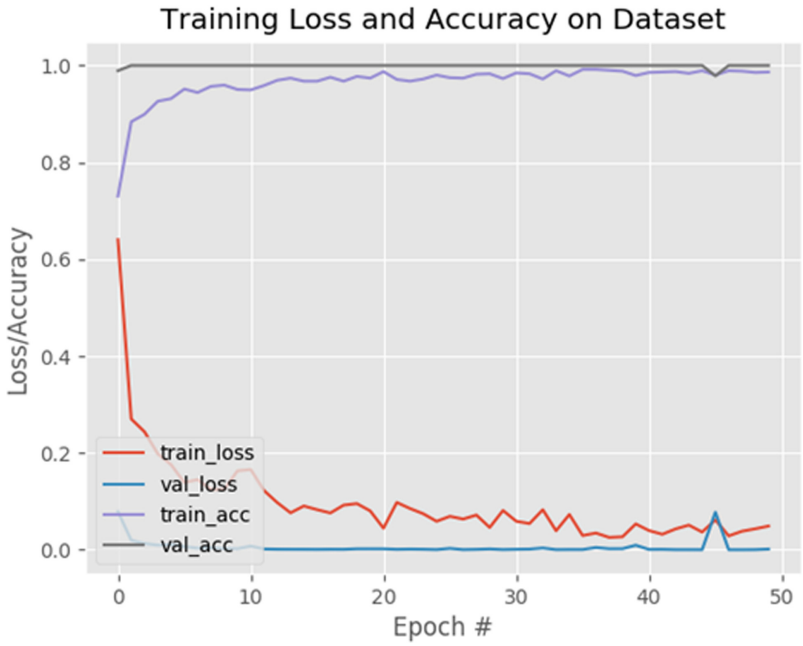


Fig. 7. Training loss and accuracy on dataset

with the following setting (rotation range = 20, the rang of the width shift = 0.2, height shift range = 0.2, zoom range = 0.15, shear range = 0.15, fill mode = "nearest", horizontal flip = True) in order to generalise well the model.Training Loss and Accuracy on training and testing datasets depicted in Fig. 7.

As can be seen from Fig. 8a and 8b, the model distinguishes real image from fake images accurately. For the convenience to differentiate, True face image is shown in blue color, while fake face image in red color. In the detection result window shown the label name, its confidence, ROI covering the face image.

4.3 Experimental Setup

In Table 3, listed experimental requirements.

Table 3. Experimental requirements

Hardware & Software requirements	Programming language	Packages
PC	Python 3.6	*Tensorflow 1.12*
Web camera		*OpenCV 3.4*
Windows OS or Linux		*Keras 2.2.4*
		scipy 1.1.0
		playsound
		pygame
		numpy
		imults
		nose
		gtts

4.4 Evaluation Metrics

For the evaluation of classification performance, the following statistical and machine learning metrics can be used: accuracy, confusion matrix, log-loss, Receiver operating characteristic (ROC) curves, precision and recall, F1-scores, and false positives per image [5]. We evaluated our approach using F1-score:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

In order to calculate the Precision and Recall, we applied our liveness detector on the 400 test real and fake face images of 20 person and counted the total number of True Positives (TPs), False Negatives (FNs), and False Positives (FPs) [13]. Precision and Recall are calculated by the equations:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

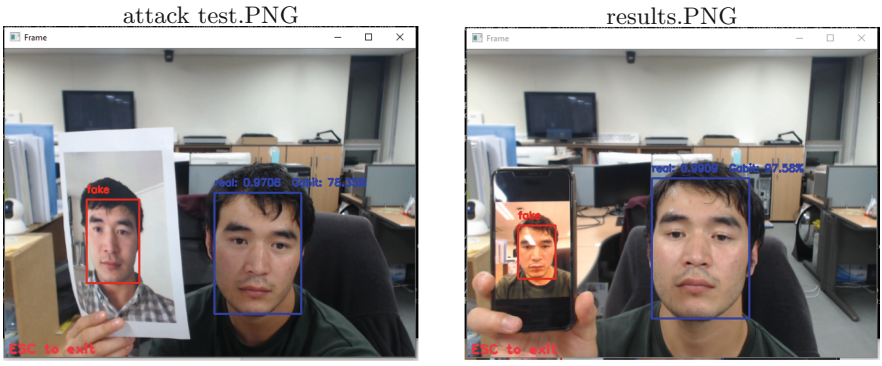
$$Recall = \frac{TP}{TP + FN} \tag{3}$$

4.5 Results

As a result of training, the accuracy of the classifier on the entire dataset was 99.8%. The experimental results obtained by applying face liveness detection model to the image stream received from a web-camera as an input image as shown in Fig. 8b. Living face images is shown with red color rectangle and recognized it with accuracy of 93.99%, whiles fake image (i.e spoofed image) displaying on smartphone is classified as a fake image and shown in a red color. From these these data, we calculated precision and recall values. After that, F1-score were calculated by Eq. 1 and added to the last column of Table 4.

5 Discussion

Our findings suggests that larger region of interest and training on various different people generalize the model. The model can distinguish new people that unseen before. The existing work in [11] cannot classify the new person's face image, while our model can distinguish correctly which is experimentally proven.



(a) Prints attack test. (b) Replay attack test.

Fig. 8. Experimental results. Liveness detection model running on PC.

Accuracy comparison between our model with the existing model is shown in the Table 4. However, the margin based liveness detection method (MLDM) has higher accuracy than the existing model (Tables 5 and 6).

Table 4. Accuracy comparison

Model	Precision	Recall	F1-score
[11]	0.709	0.675	0.6915
Our model (MLDM)	0.996	0.996	0.996

Table 5. Confusion matrix of the our trained convolutional neural network

True label	Real	117	0
	Fake	4	142
		Real	Fake
	Predicted label		

Table 6. Confusion matrix of [11]

True label	Real	64	57
	Fake	67	75
		Real	Fake
	Predicted label		

6 Conclusions

In this study, the authors presented a novel methodology for liveness detection against replay spoofing in face recognition (MLDM). We look into the dissimilar nature of imaging variability from a real face or a fake photograph face image based on the analysis of Margin based face liveness detection model, which leads to a new method to exploit the additional information contained in the given image (i.e edges and fingers). We show that phone edges and fingers also contribute to learning fake face image features, which helps to distinguish real face images and fake face images captured from smartphone display. Experiments on a real and fake face images database show that the proposed method promising replay spoofing detection performance, with advantage of real-time testing.

This is the first paper that used margin based liveness detection learning techniques to distinguish whether the given static face images are from a real live human or printed/displayed photos. In order to take measure the robustness of the proposed model, the authors are collecting more person's face images to enlarge their database to further improve and import the performance for future work.

Acknowledgment. This work was supported by Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2018-0-00245, Development of prevention technology against AI dysfunction induced by deception attack). And it was also supported by the National Research Foundation of Korea grant funded by the Korea government(MSIT) (No. 2020R1A2C1008589).

References

1. Amos, B., Ludwiczuk, B., Satyanarayanan, M.: Openface: a general-purpose face recognition library with mobile applications. Technical report CMU-CS-16-118, CMU School of Computer Science (2016)

2. Balu, G.: Resnetssd face detector, March 2018. https://github.com/gopinath-balu/computer_vision/blob/master/CAFFE_DNN/res10_300x300_ssd_iter_140000.caffemodel. Accessed 9 Sept 2020
3. Choraś, R.S.: Retina recognition for biometrics. In: Seventh International Conference on Digital Information Management (ICDIM 2012), pp. 177–180. IEEE (2012)
4. Daugman, J.: How iris recognition works. In: Bovik, A.C. (ed.) The Essential Guide to Image Processing, pp. 715–739. Elsevier, Amsterdam (2009)
5. Flach, P.A.: The geometry of roc space: understanding machine learning metrics through roc isometrics. In: Proceedings of the 20th International Conference on Machine Learning (ICML2003), pp. 194–201 (2003)
6. Hadi, A.H., Abd, Q.: Vein palm recognition model using fusion of features. Telkomnika **18**(6), 2921–2927 (2020)
7. Hoshino, S.: Personal identification authenticating with fingerprint identification, US Patent 6,636,620, 21 October 2003
8. Kim, G., Eum, S., Suhr, J.K., Kim, D.I., Park, K.R., Kim, J.: Face liveness detection based on texture and frequency analyses. In: 2012 5th IAPR International Conference on Biometrics (ICB), pp. 67–72. IEEE (2012)
9. Kim, S., Yu, S., Kim, K., Ban, Y., Lee, S.: Face liveness detection using variable focusing. In: 2013 International Conference on Biometrics (ICB), pp. 1–6. IEEE (2013)
10. Lagorio, A., Tistarelli, M., Cadoni, M., Fookes, C., Sridharan, S.: Liveness detection based on 3D face shape analysis. In: 2013 International Workshop on Biometrics and Forensics (IWBF), pp. 1–4. IEEE (2013)
11. Rosebrock, A.: Liveness Detection with OpenCV, March 2019. https://www.pyimagesearch.com/2019/03/11/liveness-detection-with-opencv/. Accessed 9 Sept 2020
12. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)
13. Seidaliyeva, U., Akhmetov, D., Ilipbayeva, L., Matson, E.T.: Real-time and accurate drone detection in a video with a static background. Sensors **20**(14), 3856 (2020)
14. Singh, A.K., Joshi, P., Nandi, G.C.: Face recognition with liveness detection using eye and mouth movement. In: 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014), pp. 592–597. IEEE (2014)