SIViP
DOI 10.1007/s11760-016-1014-2

ORIGINAL PAPER

Deep face liveness detection based on nonlinear diffusion using convolution neural network

Aziz Alotaibi[1] · **Ausif Mahmood**[1]

Received: 26 February 2016 / Revised: 11 October 2016 / Accepted: 24 October 2016

Abstract A face-spoofing attack occurs when an imposter manipulates a face recognition and verification system to gain access as a legitimate user by presenting a 2D printed image or recorded video to the face sensor. This paper presents an efficient and non-intrusive method to counter face-spoofing attacks that uses a single image to detect spoofing attacks. We apply a nonlinear diffusion based on an additive operator splitting scheme. Additionally, we propose a specialized deep convolution neural network that can extract the discriminative and high-level features of the input diffused image to differentiate between a fake face and a real face. Our proposed method is both efficient and convenient compared with the previously implemented state-of-the-art methods described in the literature review. We achieved the highest reported accuracy of 99% on the widely used NUAA dataset. In addition, we tested our method on the Replay Attack dataset which consists of 1200 short videos of both real access and spoofing attacks. An extensive experimental analysis was conducted that demonstrated better results when compared to previous static algorithms results. However, this result can be improved by applying a sparse autoencoder learning algorithm to obtain a more distinguishable diffused image.

Keywords Face liveness detection · Spoofing detection · Face detection · NUAA dataset · Replay Attack dataset anti-spoofing attacks

✉ Aziz Alotaibi
aalotaib@my.bridgeport.edu

Ausif Mahmood
Mahmood@bridgeport.edu

[1] University of Bridgeport, Bridgeport, CT 06604, USA

1 Introduction

The protection of online users' data has become an essential core function of security to create reliable, scalable, and maintainable systems. Several security methods have been developed to authenticate users' identities, including knowledge-based methods and ownership-based methods. These methods are commonly implemented in online user authentication to control access to users' data and verify their identities. However, knowledge-based methods (e.g., username/password, secret questions) are vulnerable to attacks, such as the man-in-the-middle attack, the Replay Attack, and stolen-verifier attacks. In contrast, ownership-based methods are based on something the user owns, such as a smart card or a token that can be reused, stolen, or manipulated. In both knowledge and ownership methods, the authentication system verifies what the user knows or possesses rather than truly verifying the identity of the requester [1]. As an alternative, biometrics authentication verifies the identity of requesters by using their physiological and/or behavioral characteristics. A system that employs biometric authentication ideally exhibits five qualities: robustness, distinctiveness, availability, accessibility, and acceptability [2]. In online user authentication, accessibility, and acceptability are the most significant qualities that can be found in face and voice characteristics. However, face recognition is commonly favored over other biometric traits due to its accessibility and non-intrusive form of interaction. Face recognition has been actively explored and researched in the field of security [3]. However, any photograph of a valid user (easily obtained by capturing a close-up photograph without the user's consent or obtained via the Internet) can be used to spoof face recognition systems.

A spoofing attack manipulates the system by presenting a forgery to the acquisition sensor with the goal of penetrat-

Published online: 08 November 2016

ing the biometric authentication system. More specifically, a face-spoofing attack can be accomplished by presenting a 2D image, digital video or 3D mask to the camera, mimicking the user and thus gaining access as a valid user.

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

- Applying nonlinear diffusion based on additive operator splitting (AOS) schema with a large time interval to detect edges in the input image.
- Proposing a specialized deep convolution neural network architecture that can utilize the diffused image and distinguish a real face from a fake face.
- Achieving an accuracy of 99% on the NUAA dataset, which is an improvement over all previously proposed approaches.
- Achieving a HTER of 10% on the Replay Attack dataset, which is an improvement over all previous static algorithms results.

The remainder of this paper is organized as follows. Previous related works are discussed in Sect. 2. We explained our proposed method in Sect. 3. The discussion and performance evaluation on both the NUAA and Replay Attack spoofing datasets are shown in Sect. 4. Finally, we conclude this study and discuss future work in Sect. 5.

2 Related work

Existing face liveness detection approaches can be categorized into two main groups: static approaches and dynamic approaches. Static approaches are based on an analysis of a single image and result in a non-intrusive interaction that is convenient for the majority of users. Li et al. [4] and Tan et al. [5] detected spoofing attacks based on an analysis of 2D Fourier spectra, where the structural texture of the 2D image is different from that of the 3D image. The reflections of light on 2D and 3D surfaces result in different frequency distributions. In [6], Peixoto et al. analyzed the 2D image input using a Difference of Gaussian (DoG) filter that consists of two Gaussian filters with different standard deviations to remove lighting variations and noise. Zhang et al. [7] applied multiple DoG filters to extract the high-frequency features from the input image to distinguish a fake image from a real image. Maatta et al. [8] extracted the texture of the 2D image using the multi-scale local binary pattern (LBP) to generate a concatenated histogram that was fed into a support vector machine (SVM) classifier to detect face liveness. Chingovska et al. [9] applied the LBP operator and its variation to capture the textural properties of the input image. Moreover, Kim et al. [10] calculated the diffusion speed of a single image to obtain the difference in the illumination characteristics of both 2D and 3D input images and applied a local speed pattern to extract information features that were fed into a linear SVM classifier. Yang et al. [11] introduced a component-based face recognition coding approach to extract the microtextures from twelve regions of the facial components.

Dynamic approaches are based on exploiting spatial and temporal features using a sequence of input frames, which is more computationally expensive. Some dynamic methods include "intrusive interactions," in which the user is forced to follow some instructions. Pan et al. [12] detected the eyeblinking behavior using a non-intrusive method based on a unidirectional conditional graphic framework to detect spoofing attacks. Singh et al. [13] proposed a framework to detect the spoofing attack by identifying eye and mouth movements using the Haar classifier. Kim et al. [14] presented a new novel approach utilizing the camera focus to capture variations between pixel values from two taken images. Bharadwaj et al. [15] proposed a new framework using motion magnification to detect the spoofing attack. They used the configuration LBP and motion estimation to extract the information features. Tirunagari et al. [16] applied the dynamic mode decomposition (DMD) algorithm to capture and extract dynamic visual information from the input video.

3 Proposed method

Our proposed method uses nonlinear diffusion followed by a specialized deep convolution network for face liveness detection. Nonlinear diffusion helps to distinguish a fake image from a real image by diffusing the input image quickly. Thus, the edges obtained from a flat image (picture of a picture) will fade out, whereas those from a real face will remain clear. And then, a specialized deep convolution neural network is proposed to extract the most significant features, which leads to better classification.

3.1 Nonlinear diffusion

Nonlinear diffusion is used in our face liveness detection to obtain the depth and preserve the boundary locations that help to distinguish a fake image from a real image by diffusing the input image quickly. Thus, the edges obtained from a flat image will fade out, whereas those from a real face will remain clear. In early computer vision, noise reduction and edge localization in multiscale descriptions of images has been developed and explored in the field of image processing [17]. Perona and Malik [18] proposed a nonlinear diffusion method based on a partial differential equation (PDE) [19]. They named this approach anisotropic diffusion; this approach avoids the blurring and localization issues that affect linear diffusion as follows:

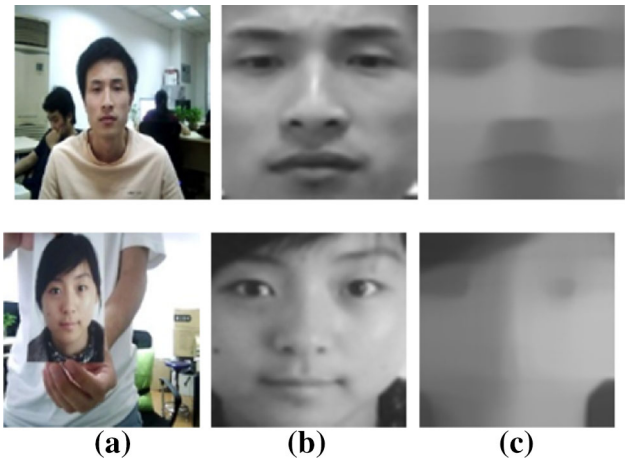

Fig. 1 Example of the NUAA database, **a** the *top image* is a real face; the *bottom image* is a fake image. **b** A normalized face with a size of 64 × 64 pixels. **c** A diffused image using AOS with a time step size of 100 and 5 iterations

$$\partial I = \partial_x (g\,(|\nabla I|)\partial_x) + \partial_y \left(g\,(|\nabla I|)\,\partial_y\right).$$

Here, the diffusivity $g(.)$ is

$$g\left(s^2\right) = \frac{1}{1 + \frac{s^2}{\lambda^2}}$$

The nonlinear diffusion filter detects the edges and preserves their locations during the diffusion process using explicit schemes. However, this schema suffers from regularization. Weickert [20] presented a semi-implicit scheme to address this problem. The scheme works with any time step size using an AOS scheme that treats the coordinates of all axes equally, as shown below:

$$(I_k)^{t+1} = \sum_{l=1}^{d} \left(dI - \tau d^2 A_l\right)^{-1} I_k^t,$$

where d denotes the input dimension and k represents the number of channels. I is the identity matrix, and A_l is the computed diffusion matrix in the vertical or horizontal direction. In 2D image, the equation would be [21]:

$$(I_k)^{t+1} = (2I - \tau 4A_x)^{-1} I_k^t + \left(2I - \tau 4A_y\right)^{-1} I_k^t.$$

The AOS scheme enables fast diffusion even with a large time step size value (e.g., 100) and can distinguish between edges in flat surfaces and rounded surfaces. As shown in Fig. 1, the edges in printed fake images fade out from the smoothing of the surface texture, whereas the real image preserves the edges and prevents the diffusion from spreading.

In order to extract the sharp the edges from the diffused image, we calculate the diffusion speed as given in [10,22]:

$$I\,(x, y) = \left|\log\left(I^0\,(x, y) + 1\right) - \log\left(I^l\,(x, y) + 1\right)\right|$$

where I^0 represents the original image and I^l denotes the diffused image. As shown in Fig. 1, the real image surface has relatively sharp edges (e.g., nose and cheek). In contrast, the surface of the fake images has smoother edges. All previous approaches used handcrafted features, such as the LBP, to extract the information features. That approach has some limitations, such as a limited ability to extract complex features. Therefore, this work uses deep learning with gradient descent to extract the discriminative and higher-level features from the diffused image. We propose a specialized deep convolution neural network architecture to extract the most significant features, which leads to better classification, as explained in the next section.

3.2 Convolution neural network

Deep learning algorithms have been successfully applied in several vision tasks such as object detection [23,24], handwriting recognition [25], face detection [26] and face recognition [27]. The convolution neural network (CNN) was first introduced by LeCun et al. [24,25] and is predominantly a biologically inspired hierarchical multilayered neural network approach that simulates the human visual cortex and detects translation invariance features [28]. CNNs are designed to extract the local features by combining three architectural concepts that perform some degree of shift, scale, distortion invariance, local receptive fields, shared weight and subsampling. The ability of both convolution layers and subsampling layers to learn distinctive features from the diffused image helps in extracting features and achieving the best classification for face liveness detection.

3.2.1 Our specialized convolution neural network architecture

The proposed CNN was trained using the standard back-propagation algorithm, as shown in Table 1. We trained our network using the stochastic gradient descent method to calculate the true gradient at each iteration, which is considered faster than batch learning. CNNs learn faster from the unexpected input; thus, we shuffle the data randomly at each iteration. The value of the input image pixels is normalized between zero and one, setting the mean close to zero and the variance close to one. We used the hyperbolic tangent (Tan$h$) as an activation function. We randomly initialized the weights (w), and biases (b) are between 1 and −1. Also, a small learning rate with a value of 0.005 is used. In the last layer, the softmax activation function is used as a classifier to approximate the expected output to between 0 and 1 in our binary classification. Our specialized deep convolution neural network consists of five convolutional and subsampling layers, and the last layer is the fully connected layer, as shown in Fig. 2. The input image has a size of 64 × 64

Table 1 Forward and backpropagation algorithm

Algorithm 1. Forward and backpropagation algorithms for our proposed convolution neural network

All weights (w) and biases (b) are initialized to a value between -1 and 1, and the learning rate λ is set to 0.005
Input (I) of size 64 × 64
For i =1 to I **do**
Forward
For layers l=1 to L **do**
For FeatureMap f =1 to F **do**
If layer l is C layer **then**

$$i_f^{(l)}(x,y) = \emptyset\left(\left(\sum_{k\in K}\sum_{(m,n)} w_f^{(l)}(m,n).\, i_k^{(l-1)}(x+m, y+n)\right) + b_f^{(l)}(x,y)\right)$$

Else If layer l is S layer **then**

$$i_f^{(l)}(x,y) = \emptyset\left(\left(w_f^{(l)} * \sum_{(m,n)} i_f^{(l)}(2x+m, 2y+n)\right) + b_f^{(l)}(x,y)\right)$$

Else If layer is fully connected **then**

$$i_f^{(l)} = \emptyset\left(\left(\sum_{k=1}^{K}(w_{kf}^{(l)}.\, i_{kf}^{(l-1)})\right) + b_f^{(l)}\right)$$

End if
End for
End for
Backpropagation
For layer l=L-1 to 1 **do**
If layer l= L **then**

$$\delta_k^{(l)} = (O_k^{(l)} - E_k^{(l)})$$

Else if layer l+1 is fully connected **then**

$$\delta_k^{(l)}(x,y) = \left(\sum_{k=1}^{K}\sum_{(x,y)} \delta_k^{(l+1)} w_{kf}^{(l+1)}(x,y)\right) \emptyset'(A_{(k)}^{(l)})$$

Else if layer l+1 is C **then**

$$\delta_f^{(l)}(x,y) = \left(\sum_{k=K_{l\,(Convolution)}}\sum_{(m+i,n+j)} \delta_k^{(l+1)}(i,j)\, w_k^{(l+1)}(m,n)\right) \emptyset'(A_{(k)}^{(l)})$$

Else if layer l+1 is S **then**

$$\delta_f^{(l)}(x,y) = \left(\delta_k^{(l+1)}(2x+m, 2y+n)/\, w_f^{(l+1)}(m,n)\right)\emptyset'(A_{(k)}^{(l)})$$

End if
End for
For layer l=1 to L **do**
If layer l is C **then**

$$w_f^{(l)}(m,n) = w_f^{(old)}(m,n) + \left(-\lambda \sum_{k\in K}\sum_{(x,y)}\left(\delta_f^{(l)}(x,y)\, i_k^{(l-1)}(x+m, y+n)\right)\right)$$

Else If layer l is S **then**

$$w_f^{(l)}(m,n) = w_f^{(l)}(m,n)$$

Else If layer is fully connected**, then**

$$w_f^{(l)}(m,n) = w_f^{(old)}(m,n) + (-\lambda \sum_{k\in K} i_k^{(l-1)}.\, \delta_f^{(l)})$$

End if
If layer is C **then**

$$b_f^{(l)}(x,y) = b_f^{(old)}(x,y) + (-\lambda.\, \delta_f^{(l)}(x,y))$$

Else If layer is S **then**

$$b_f^{(l)}(x,y) = b_f^{(old)}(x,y) + (-\lambda.\, \delta_f^{(l)}(x,y))$$

Else if layer is fully connected **then**

$$b_f^{(l)}(x,y) = b_f^{(old)}(x,y) + (-\lambda.\, \delta_f^{(l)}(x,y))$$

End if
End for
Until Convergence

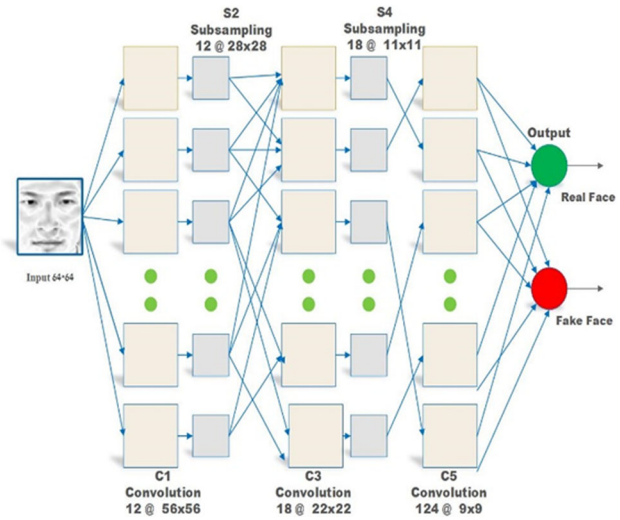

Fig. 2 Our proposed convolution neural network architecture for NUAA dataset

pixels. Layer C1 is a convolution layer with 12 feature maps. Each unit in the feature map is a result of connecting a 9 × 9 neighbor in the input image. The size of the feature map in C1 is 56 × 56 pixels. Layer S2 is a subsampling layer with 12 feature maps of 28 × 28 pixels. Each feature map in the subsampling layer is connected to an average kernel 2 × 2 neighborhood from the previous corresponding feature map in C1. The average 2 × 2 kernel is non-overlapping. Therefore, the size of the feature map in S2 is half the size of the feature map in C1. C3 is a convolution layer with 18 feature maps of 22 × 22 pixels. Each feature map takes inputs from 4 random feature maps from the previous S2 subsampling layer. All 4 feature maps from subsampling are connected to only one 7 × 7 kernel. Layer S4 is a subsampling layer with 18 feature maps of 11 × 11 pixels. Each feature map in the subsampling is connected to an average 2 × 2 kernel neighborhood from the previous corresponding feature map in C3. C5 consists of 124 feature maps of 9 × 9 pixels. Each unit in the feature map is a result of connecting the 3 × 3 neighbor in the input. Each feature map takes 1 random feature map from the previous S4 subsampling layer. Finally, the last layer is the output layer, a fully connected layer.

4 Discussion and performance evaluation

The goal of this section is to demonstrate the effectiveness of our proposed deep convolution neural network, which utilizes a signal-diffused image to differentiate between a real face and a fake face. The nonlinear diffusion based on an AOS scheme allows us to apply a large time step to speed up the diffusing process and to distinguish the edges and surface in the input image. To the best of our knowledge, our

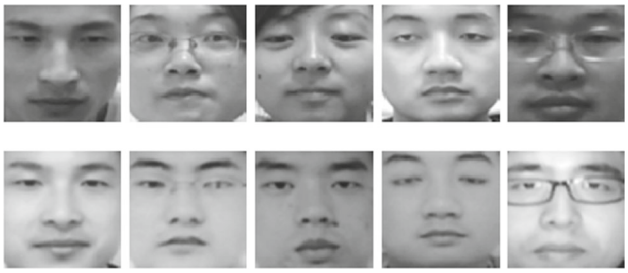

Fig. 3 Example of the NUAA database (*top* live photograph, *bottom* fake photograph

Fig. 4 Examples of the Replay Attack database

results using the NUAA are state of the art compared with all previous methods. We divide this section into six parts. First, we introduce a widely used benchmark dataset called the NUAA dataset and Replay Attack database. Then, we present the analysis of our proposed deep convolution neural network method. Moreover, we evaluate the performance of our approach using the NUAA and Replay Attack dataset and compare it with other previously proposed techniques. Finally, we provide the computational time required by our proposed method.

4.1 NUAA dataset

The NUAA Photograph Imposter Database [5], released in 2010, is publicly available and widely used for evaluating static face liveness detection. The NUAA database consists of 12,614 images of both live and photographed faces. The database images consist of 15 subjects. The database images were converted into a grayscale representation and resized to 64×64 pixels, as shown in Fig. 3. The database is divided into a training set with a total of 3491 images and a test set with a total of 9123 images. There is no overlap between the training and test sets.

4.2 Replay Attack database

Replay Attack Database [9] was released in 2012 and is publicly available and widely used. It consists of 1200 short videos of 50 different subjects divided into 200 real-access videos and 1000 spoofing attack videos. The Replay Attack database is divided into three subsets: training, development and testing (Fig. 4).

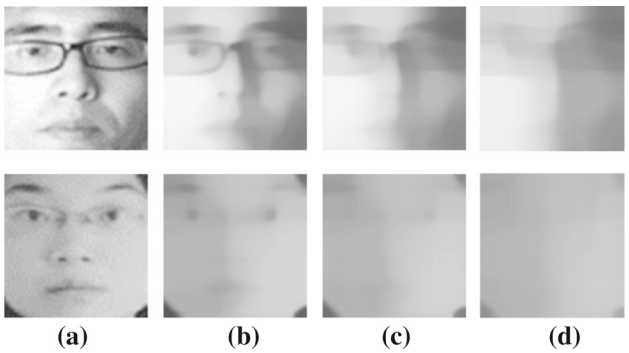

Fig. 5 **a** Normalized image, **b** diffused image with $\tau = 50$, **c** diffused image with $\tau = 100$, and **d** diffused image with $\tau = 200$

4.3 Discussion and analysis

In this subsection, we discuss and analyze the efficiency and robustness of our approach for detecting face-spoofing attacks utilizing only one input image. In the NUAA dataset, the input image is normalized to 64×64 pixels. This normalized image has no background, which reduces the time required for processing—particularly when passing the input image through our deep convolution neural network. Some dynamic techniques utilize the background to extract features that increase the time required for processing; however, our approach focuses on extracting sharp edges from the input surface rather than detecting other features from the background. We apply the AOS-based diffusion scheme to extract sharp edges and surface textures, such as the nose, eyes, lips and cheek. These characteristics form most edges and textures in faces that can help distinguish 2D from 3D images when applying a large time step value. Re-capturing the input image twice destroys the depth information and changes the pixel locations. After conducting many experiments with different time step values, we found out that a time step of ($\tau = 100$) and iterating five times ($L = 5$) yields the best result, as shown in Table 4. Using a larger time step (one greater than $\tau = 100$) causes the most important features, such as the edges and location, to fade out, as shown in Fig. 5. Also, Increasing the number of iterations from 5 to 10 requires additional time and blurs the face image, as shown in Fig. 7. The iteration $L = 5$ yields an accuracy of 99% on NUAA dataset, whereas iterations of $L = 10$ and $L = 20$ yield accuracy rates of 93 and 92%, respectively. Our proposed feature extraction, CNN, has proven to be powerful in extracting not only the depth information but also the texture surface of the faces, as shown in Fig. 6. The trained kernels are able to distinguish the speed-diffused images. After visualizing the output of the first convolution layer, we found out that real face has more edges and distinct corners around the eye, nose, lips and cheek regions, where the fake face has fewer edges around the eyes only.

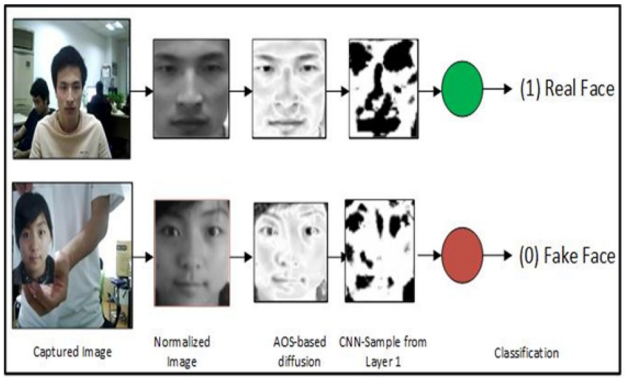

Fig. 6 Process of our proposed approach

Table 2 Performance evaluation for different numbers of iterations using the Replay Attack dataset

τ	L	Accuracy	τ	L	Accuracy
100	1	16.5	100	6	11.875
100	2	13.5	100	7	11.875
100	3	15.75	100	8	12.625
100	4	14.875	100	9	11.00
100	5	10.00	100	10	14.625

The time step is fixed at a value of 100

Table 3 HTER (%) of classification for the Replay Attack dataset

Methods	Test (%)
$\text{LBP}^{u2}_{3\times3} + x^2$ [9]	34.01
$\text{LBP}^{u2}_{3\times3}$ + LDA [9]	17.17
$\text{LBP}^{u2}_{3\times3}$ + SVM [9]	15.16
LBP + SVM [8]	13.87
DS-local speed pattern [10]	12.50
Our proposed approach	10

4.4 Performance evaluation using the Replay Attack dataset

To evaluate the performance of our approach, we conducted many experiments with different time step size values (τ) and different iteration numbers (L) using the videos in the Replay Attack dataset, as shown in Table 2. We computed the half total error rate (HTER) to measure the performance of our proposed approach [29]. The HTER is half of the sum of the false rejection rate (FRR) and false acceptance rate (FAR), as shown below:

$$\text{HTER} = \frac{\text{FRR} + \text{FAR}}{2}$$

Table 4 Performance with different parameters using the NUAA dataset

τ	L	Accuracy	τ	L	Accuracy
40	5	90.23	40	10	97.56
60	5	94.03	60	10	93.29
80	5	93.31	80	10	97.36
100	5	**98.99**	100	10	95.99
120	5	98.21	120	10	93.97
140	5	97.96	140	10	94.11
160	5	97.15	160	10	97.01
180	5	96.74	180	10	94.74
200	5	94.10	200	10	94.60

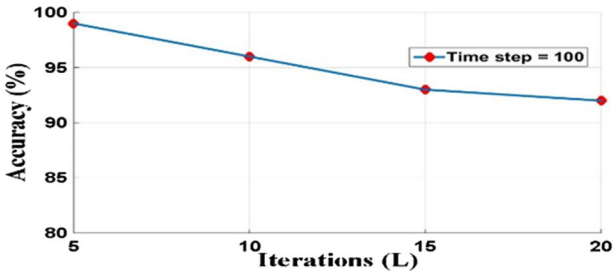

Fig. 7 Performance evaluation for different numbers of iterations for NUAA dataset. The time step is fixed at a value of 100

Table 3 provides a summary of the HETR, showing the classification result of our proposed approach compared with different approaches.

4.5 Performance evaluation using the NUAA dataset

In this subsection, we evaluate the performance of our approach; we conducted many experiments with different time step size values (τ) and different iteration numbers (L) using the images in the NUAA dataset, as shown in Table 4. The best detection accuracy achieved using the NUAA dataset was 99% using values of $\tau = 100$ and $L = 5$. Using a larger time step value and a larger number of iterations does not always yield higher accuracy, as shown in Table 4. For example, experiments where $\tau = 120$ and $L = 5$ resulted in an accuracy of 98.21%, and experiments where $\tau = 120$ and $L = 10$ resulted in an accuracy of 93.97%. Thus, increasing the iteration number not only fails to improve the accuracy rate of our proposed deep convolution neural network, as shown in Fig. 7, but also requires more computational time.

To prove the efficiency and effectiveness of our approach, we compared the performance of our proposed deep convolution neural network using the NUAA dataset with all previously proposed approaches. The compared approaches were: multiple difference of Gaussian (M-DoG) [7], high

Table 5 Performance comparison using the NUAA dataset

Methods	Accuracy (%)
M-DoG [7]	81.8
HDF [4]	84.5
DoG-LRBLR [5]	87.5
M-LBP [8]	92.7
DoG-S L [6]	94.5
CDD [11]	97.7
DS-LSP [10]	98.5
Our proposed approach	99.0

Fig. 8 Examples of misclassified face images. The *top face images* are rejected clients, and the *bottom face images* are accepted printed images

descriptor frequency [4], DoG-sparse low-rank bilinear logistic regression (DoG-LRBLR) [5], multiple local binary pattern (M-LBP) [8], DoG-sparse logistic (DoG-SL) [6], component-dependent descriptor (CDD) [11] and the diffused speed-local speed pattern (DS-LSP) [10]. As shown in Table 5, our proposed approach achieves the best performance with an accuracy of 99%.

We also computed the half total error rate (HTER) to measure the performance of our proposed approach using NUAA dataset.

Table 6 provides a summary of the HETR, showing the classification result of our proposed approach compared with different approaches. The FRR is 0.47%, and the FAR is 1.31%. Our HTER is 0.98%. Analysis of the misclassified face images indicated that over-exposures, blurring and reflections affected our proposed approach's ability to detect spoofing attacks, as shown in Fig. 8.

4.6 Processing time

In this subsection, we analyze the computational time required by our method in further detail. We divided the processing of our approach to detect spoofing attacks into three steps: the diffusion process, CNN-based feature extraction and classification. The total time required for the

Table 6 HTER (%) of classification for the NUAA dataset

Method	HTER (%)
Our proposed CNN	0.98
$LBP^{u2}_{3\times3}$ + LDA	17.08
$LBP^{u2}_{3\times3}$ + SVM	19.03
LBP [8] + SVM	13.17

Table 7 Time processing

Diffusion process	Our CNN	Classification	Total
0.023	0.052	0.001	0.076 (per s)

framework of the proposed approach to detect the spoofing attack is approximately 0.076 s/person, as shown in Table 7. The proposed approach was implemented on a PC with an Intel® Core™ i7-4500U CPU running at 1.80 GHz and 8 GB RAM without parallel processing. The application was written using Visual Studio 2013 and the C# language. As shown in Table 7, the feature extraction using the convolution neural network consumes the bulk of the detection time (Table 7).

5 Conclusions and future work

In this paper, an effective and robust approach was proposed to address the problem of face-spoofing attacks using a single image. We used an AOS-based schema with a large time step size to obtain the speed-diffused image. All the previous approaches used hand-designed features extraction, such as the Difference of Gaussian and the LBP algorithm, to extract the information features from the input image. In contrast, this work uses a deep learning algorithm with gradient descent. In this study, we conducted numerous experiments with different parameters to demonstrate the performance of our proposed approach. The best-performing classification result on NUAA dataset was an accuracy of 99% when applying a time step of ($\tau = 100$) and setting the number of iterations to ($L = 5$). The experiment shows that using a large time step destroys the edges and changes the pixel locations. Furthermore, increasing the number of iterations leads to reduced accuracy. Future work will investigate whether implementing deep learning especially the sparse autoencoder learning algorithm to obtain the diffused image will help in extracting the most significant features compared with the AOS-based scheme. The diffused image obtained using autoencoder will be fed to the deep convolution neural network.

Compliance with ethical standards

Conflict of interest We have no conflicts of interest to disclose.

References

1. Ren, X., Wu, X.W.: A novel dynamic user authentication scheme. In: 2012 International Symposium on Communications and Information Technologies (ISCIT), pp. 713–717 (2012)
2. Wayman, J., Jain, A., Maltoni, D., Maio, D.: An Introduction to Biometric Authentication Systems. Springer, Berlin (2005)
3. De Marsico, M., Nappi, M., Riccio, D., Tortora, G.: Entropy-based template analysis in face biometric identification systems. Signal Image Video Process. **7**, 493–505 (2013)
4. Li, J., Wang, Y., Tan, T., Jain, A.K.: Live face detection based on the analysis of fourier spectra. In: Defense and Security. International Society for Optics and Photonics, pp. 296–303 (2004)
5. Tan, X., Li, Y., Liu, J., Jiang, L.: Face liveness detection from a single image with sparse low rank bilinear discriminative model. In: Computer Vision–ECCV 2010, pp. 504–517. Springer, Berlin (2010)
6. Peixoto, B., Michelassi, C., Rocha, A.: Face liveness detection under bad illumination conditions. In: 2011 18th IEEE International Conference on Image Processing (ICIP), pp. 3557–3560 (2011)
7. Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D., Li, S.Z.: A face antispoofing database with diverse attacks. In: 2012 5th IAPR International Conference on Biometrics (ICB), pp. 26–31 (2012)
8. Maatta, J., Hadid, A., Pietikainen, M.: Face spoofing detection from single images using micro-texture analysis. In: International Joint Conference on Biometrics (IJCB), pp. 1–7 (2011)
9. Chingovska, I., Anjos, A., Marcel, S.: On the effectiveness of local binary patterns in face anti-spoofing. In: BIOSIG—Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1–7 (2012)
10. Kim, W., Suh, S., Han, J.J.: Face liveness detection from a single image via diffusion speed model. IEEE Trans. Image Process. **24**, 2456–2465 (2015)
11. Jianwei, Y., Zhen, L., Shengcai, L., Li, S.Z.: Face liveness detection with component dependent descriptor. In: International Conference on Biometrics (ICB), pp. 1–6 (2013)
12. Pan, G., Sun, L., Wu, Z., Lao, S.: Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In: IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007, pp. 1–8 (2007)
13. Singh, A.K., Joshi, P., Nandi, G.C.: Face recognition with liveness detection using eye and mouth movement. In: International Conference on Signal Propagation and Computer Technology (ICSPCT), pp. 592–597 (2014)
14. Kim, S., Yu, S., Kim, K., Ban, Y., Lee, S. : Face liveness detection using variable focusing. In: International Conference on Biometrics (ICB), pp. 1–6 (2013)
15. Bharadwaj, S., Dhamecha, T.I., Vatsa, M., Singh, R.: Computationally efficient face spoofing detection with motion magnification. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 105–110 (2013)
16. Tirunagari, S., Poh, N., Windridge, D., Iorliam, A., Suki, N., Ho, A.T.S.: Detection of Face Spoofing Using Visual Dynamics. IEEE Trans. Inf. Forensics Secur. **10**, 762–777 (2015)
17. Witkin, A.P.: Scale-space filtering. ed: Google Patents (1987)
18. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. **12**, 629–639 (1990)
19. Nadernejad, E., Sharifzadeh, S., Forchhammer, S.: Using anisotropic diffusion equations in pixon domain for image denoising. Signal Image Video Process. **7**, 1113–1124 (2013)
20. Weickert, J., Romeny, B.T.H., Viergever, M.: Efficient and reliable schemes for nonlinear diffusion filtering. IEEE Trans. Image Process. **7**, 398–410 (1998)
21. Ralli, J.: PDE based image diffusion and AOS (2014). http://jarnoralli.com/images/pdf/non_linear_image_diffusion_and_aos_ralli_2014.pdf
22. Land, E.H., McCann, J.: Lightness and retinex theory. JOSA **61**, 1–11 (1971)
23. Jia, B., Feng, W., Zhu, M.: Obstacle detection in single images with deep neural networks. Signal Image Video Process. **10**, 1–8 (2015)
24. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**, 2278–2324 (1998)
25. Le Cun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., Handwritten digit recognition with a back-propagation network. In: Advances in Neural Information Processing Systems, pp. 396–404. Morgan Kaufmann Publishers Inc. San Francisco, CA (1990)
26. Garcia, C., Delakis, M.: Convolutional face finder: a neural architecture for fast and robust face detection. IEEE Trans. Pattern Anal. Mach. Intell. **26**, 1408–1423 (2004)
27. Lawrence, S., Giles, C.L.: Ah Chung, T., Back, A.D.: Face recognition: a convolutional neural-network approach. IEEE Trans. Neural Netw. **8**, 98–113 (1997)
28. Fasel, B.: Robust face analysis using convolutional neural networks. In: Proceedings. 16th International Conference on Pattern Recognition, pp. 40–43 (2002)
29. Bengio, S., Mariéthoz, J.: A statistical significance test for person authentication. In: ODYSSEY04—The Speaker and Language Recognition Workshop (2004)