Expert Systems With Applications 108 (2018) 159–169

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

LiveNet: Improving features generalization for face liveness detection using convolution neural networks

Yasar Abbas Ur Rehman*, Lai Man Po, Mengyang Liu

Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong SAR, China

ARTICLE INFO

Article history:
Received 8 January 2018
Revised 24 March 2018
Accepted 5 May 2018
Available online 8 May 2018

Keywords:
Convolution neural networks
Face anti-spoofing
Face liveness detection
Face-biometric
VGG-11
Bootstrapping
EER
HTER

ABSTRACT

Performance of face liveness detection algorithms in cross-database face liveness detection tests is one of the key issues in face-biometric based systems. Recently, Convolution Neural Networks (CNN) classifiers have shown remarkable performance in intra-database face liveness detection tests. However, a little effort has been made to improve the generalization capability of CNN classifiers for cross-database and unconstrained face liveness detection tests. In this paper, we propose an efficient strategy for training deep CNN classifiers for face liveness detection task. We utilize continuous data-randomization (like bootstrapping) in the form of small mini-batches during training CNN classifiers on small scale face anti-spoofing database. Experimental results revealed that the proposed approach reduces the training time by 18.39%, while significantly lowering the HTER by 8.28% and 14.14% in cross-database tests on CASIA-FASD and Replay-Attack database respectively as compared to state-of-the-art approaches. Additionally, the proposed approach achieves satisfactory results on intra-database and cross-database face liveness detection tests, claiming a good generality over other state-of-the-art face anti-spoofing approaches.

1. Introduction

In recent years, research in face, fingerprint, iris and palm based biometric has attained significant interest from biometric communities (Choudhury, Then, Issac, Raman, & Haldar, 2018; Galbally, Alonso-Fernandez, Fierrez, & Ortega-Garcia, 2012; Nguyen, Fookes, Jillela, Sridharan, & Ross, 2017; Sajjad et al., 2018). With convenient use and almost no physical interaction with the biometric devices, face recognition systems and face authentication systems have been widely used in portable electronic devices such as cell phones, laptops, tablets, and in non-portable electronic devices such as attendance registration systems in colleges and universities, surveillance systems at the airport and other sensitive areas (Ramachandra & Busch, 2017). Despite their widespread use in today's modern electronic systems, face recognition systems and face authentication systems like other biometric traits are vulnerable to face spoofing attacks, i.e. an intruder can easily fool the face authentication system by presenting it a forged face of a genuine user. In addition, there are multiple levels of forged face attacks which ranges from single photographic image attack to 3D face masks attacks (Boulkenafet, Akhtar, Feng, & Hadid, 2017). Thus, a robust face authentication system and face verification system should distinguish between any type of forged face and a genuine face, which make face anti-spoofing techniques, like other state-of-the-art biometric anti-spoofing techniques, as a baseline requirement in modern electronic face authentication systems and face verification systems design.

A robust face anti-spoofing system generally deals with multiple types of face spoofing attacks. Realizing the need to make a face recognition systems fool proof, the face-biometric communities have developed and made publically available several state-of-the-art face anti-spoofing datasets and systems to aid the development of robust face anti-spoofing techniques (Marcel, 2013). These state-of-the-art face anti-spoofing datasets have multiple definition of face spoofing attacks in accordance with the medium used to trick the face authentication systems. In addition, to assess and analyze the performance of the face anti-spoofing algorithms, various protocols have been defined as well, that include various tests such as fake face test, quality test etc.

Many efficient face anti-spoofing algorithms have been developed in the recent decade, which can be broadly classified into fixed feature based face anti-spoofing algorithms (Menotti et al., 2015) and automatic learnable feature based face anti-spoofing algorithms (Feng, Po, Li, Xu et al., 2016). The fixed feature based face anti-spoofing algorithms utilizes hand-crafted features of genuine face and spoofed face for face anti-spoofing applications. Usually, the features of genuine faces and fake faces are computed prior to training an algorithm for face anti-spoofing applications. Fixed

* Corresponding author.
E-mail addresses: yaurehman2-c@my.cityu.edu.hk (Y.A.U. Rehman), eelmpo@cityu.edu.hk (L.M. Po), mengyaliu7-c@my.cityu.edu.hk (M. Liu).

https://doi.org/10.1016/j.eswa.2018.05.004
0957-4174/

feature based algorithms are further broadly classified into motion based, texture based, image quality based, 3D shape based and techniques that exploit multi-spectral reflectance. On the other hand, automatic learnable feature based face anti-spoofing algorithms do not use hand-crafted features but use deep learning techniques, such as Convolution Neural Networks (CNNs) for classification of genuine faces and spoofed faces. A CNN network learn the features of genuine faces and fake faces during its training phase. A CNN network basically maps the raw input pixels of an image to the output probability by passing the input image through intermediate hidden layers. The number of hidden layers determine the depth of a CNN network. Practically, a deep CNN network is the choice for many applications, however small training data, especially for face anti-spoofing systems, has restricted its use in face anti-spoofing applications.

Till now, there have been very few methods, reported in the recent years, that use deep CNN networks for face anti-spoofing application. Although, obtaining good accuracy on face spoofing detection; most of these methods have used a shallow and pre-trained Alex Net (Krizhevsky, Sutskever, & Hinton, 2012) along with transfer learning techniques for face anti-spoofing application development, which limits its implementation in real-time face anti-spoofing systems. Additionally, in these methods, the end-to-end learning strategy has not been utilized, and traditional feature extraction methods have been used prior to training a CNN network, and an SVM classifier after training a CNN network. Further these face anti-spoofing algorithms have been tested on intra-database face spoofing detection tests and very few articles established cross-database face spoofing detection tests.

Acknowledging the remarkable success of deep CNN networks in image classification and object detection since its successful incarnation in 2012 (Krizhevsky et al., 2012), this paper proposes a deep CNN network for face anti-spoofing application. The main contributions of this paper are as follow:

1. First, an efficient strategy is presented for training CNN networks on face anti-spoofing datasets that have limited training samples. The proposed training strategy in this paper is a data randomization technique which is similar to bootstrapping. This provide an efficient mechanism for training deep CNN networks, in which deep CNN networks are trained effectively, using end-to-end learning, on databases with limited training samples like face anti-spoofing databases. Specifically, it helps to improve the generalization capability of CNN networks in classifying unknown types of attacks and further reduces the training time substantially.
2. Second, detail analyses of the performance of proposed training strategy for CNN networks on intra-database face anti-spoofing tests and cross-database face anti-spoofing tests are presented that include spoofing or liveness detection under various protocols tests respectively.

The rest of this paper is organized as follows: Section 2 presents a review of state-of-the-art face anti-spoofing algorithms. Section 3 describes the details of the proposed CNN networks, the proposed strategy for training CNN networks for face anti-spoofing application and the metrics used for evaluation of the proposed CNN networks for face anti-spoofing applications. Section 4 provide a detail evaluation of the performance of CNN networks on intra-database tests and cross-database tests. Section 5 provides a discussion on the proposed method and experimental work. Finally, Section 6 provides concluding remarks and future work.

2. Conventional CNN network based face liveness detection

Owing to remarkable success of CNN networks in various categories of image classification and object detection, the CNN networks are now being used for face detection (Li, Lin, Shen, Brandt, & Hua, 2015) and face recognition (Schroff, Kalenichenko, & Philbin, 2015). Face anti-spoofing is a special case of intra-class face recognition, in which the aim is to recognize a face image as a genuine or spoofed. Face anti-spoofing algorithms can be divided into two broad categories, face liveness detection and spoofing attack classification. Face liveness detection is fundamentally a binary classification problem, in which the aim is to classify a face as genuine or spoofed. On the other hand, in spoofing attack classification, the task is to not only classify a face image as either genuine or spoofed but also to classify the type of spoofing attack in case of a spoofed face image. A multitude of literature is available for fixed feature based face anti-spoofing algorithms (Galbally, Marcel, & Fierrez, 2014; Määttä, Hadid, & Pietikäinen, 2011; 2012; Waris, Zhang, Ahmad, Kiranyaz, & Gabbouj, 2013), which are broadly classified as motion based, texture based, image quality based and methods that used 3D and spectral reflectance properties. However, little literature is available for face anti-spoofing techniques that utilized deep learning algorithms such as CNN networks. In the following paragraph, a review of the-state-of-the-art techniques is presented that use deep CNN networks for face anti-spoofing application.

In Menotti et al. (2015), a CNN network was used for combined face, iris and finger-print spoofing detection. Although, a high accuracy and low HTER was reported on face anti-spoofing datasets for face liveness detection, there were no results reported on cross-database face liveness detection which might question the generality of the proposed method in real-time applications. Further, they proposed a very shallow CNN network for face liveness detection that was unable to capture the abstract and high level feature maps as in deep CNN networks. In Xu, Li, and Deng (2015), the authors introduced a 2 layers CNN network with a single LSTM layer for face liveness detection application. The CNN network was used to capture the 2-dimensional feature maps and the LSTM network was used to capture the temporal information. Although, the proposed model provided satisfactory results on face liveness detection problem using CASIA-FASD database (Zhang et al., 2012), it was unable to provide any accuracy of classifying multiple types of attacks. Further, there was no information about the number of samples of each subject used for training CNN network, which is a key element in training of CNN networks. Further, the CNN network used was quite shallow, 2 convolution layers were used which may not perform well on multi-class face anti-spoofing problem.

In Alotaibi and Mahmood (2017), the authors proposed a shallow CNN network for the classification of spoofed face and real face. Their method utilized the non-linear diffusion based on additive operator splitting schema to get a diffused image that was later fed to a CNN network to classify an input face image as real or fake. However, their approach has certain limitations. First, they used a very shallow network consisting of only 3 convolution layers which could provide less generalization capability as the initial layers capture the low-level information. The use of diffusion approach to capture the gradient information was redundant because the early stages of CNN networks performed the same process. Second, their approach cannot be considered as an end-to-end learning approach as hand-crafted features were first obtained prior to training a CNN network. Most importantly, the authors did not provide any results on cross-database testing which question the generality of the proposed approach. In Yang, Lei, and Li (2014), the authors proposed a CNN network for face anti-spoofing problem to classify various attacks on two state-of-the-art face anti-spoofing datasets, i.e. CASIA-FASD database and Replay-Attack database (Chingovska, Anjos, & Marcel, 2012). In their method, a face region was first localized followed by data augmentation at five different scales before training a CNN net-

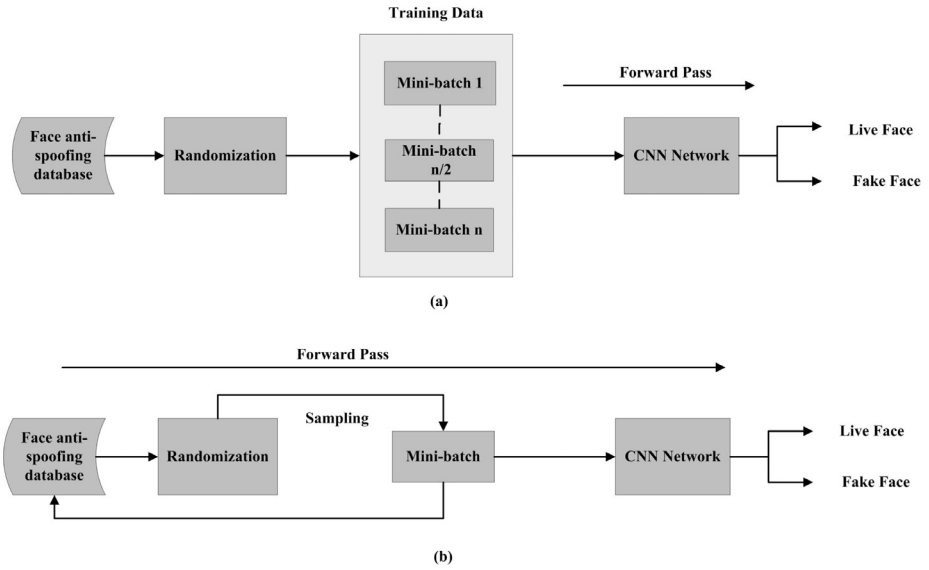

Fig. 1. The sampling is done in the form of mini-batches. (a) Conventional method for training CNN Networks. (b) Proposed method for training CNN networks.

work. However, after training CNN networks, the final features were used to train the Support Vector Machine (SVM) classifier for face anti-spoofing. Even their proposed method obtained remarkable results on CASIA-FASD and Replay-Attack database, it cannot be regarded as an end-to-end learning. The approach in Li et al. (2016) exploited the layer-wise features of CNN network to train an SVM algorithm. In their CNN network, they used various layers of shallow to deep VGG networks features for face liveness detection. However, they used a pre-trained network for face anti-spoofing and thus there are no details of training a CNN network. In Feng, Po, Li, and Yuan (2016), the authors proposed a combination of shearlet-based features and stacked auto-encoders for the face liveness detection and face spoofing attack classification. In Manjani, Tariyal, Vatsa, Singh, and Majumdar (2017) a deep dictionary approach has been proposed for face liveness detection application. However, no results were reported for intra-database and cross-database spoofing attack classification.

The approaches mentioned above were carefully designed for binary classification problem, however their performance on cross-database tests has been very low. Further, the inherent property of deep learning algorithms, i.e. end-to-end learning was not fully utilized in these algorithms. Thus, this paper aims to present an efficient approach that utilizes the end-to-end learning property of CNN networks for face liveness detection. The next section presents the proposed methodology adopted to achieve this objective.

3. Methodology

The main motivation behind using CNN network for face anti-spoofing is to capture the discriminative and generalized feature maps from face images that can help in identifying face from non-face. To learn generalized feature maps, a CNN network must be trained effectively without any over-fitting. Unfortunately, the conventional deep CNN training techniques over-fit when trained on limited scale face anti-spoofing databases. Fig. 1 shows the conventional and the proposed method respectively for training CNN networks. As can be observed in Fig. 1(a), in conventional method, the data is randomize only once prior to training CNN network. However, in the proposed approach, the training data is continuously randomized before applying to CNN network in a form of single small mini-batches as shown in Fig. 1(b). This proposed training strategy greatly circumvent the effects of over-fitting in deep CNN networks caused by low amount of training data, which is explained in the following sections.

3.1. Training with continuous data-randomization

We utilize data randomization technique that is similar to bootstrapping. Rather than randomly arranging the training set once, we continuously pick random mini-batches from the whole training set at each training epoch. Let suppose the total number of epochs, for which the network to be trained, is represented by E_T and the sub-epochs is represented by ϵ_{ps}. Each K complete epoch is represented by E_K, where K is an integer. The training mini-batch can be represented by $B_s(x_i^t)$. Then

$$E_k = \epsilon_{ps} \times B_s rand(x_i^t), \tag{1}$$

$$E_T = K \times \epsilon_{ps} \times B_s rand(x_i^t), \tag{2}$$

$$E_T = C \times B_s rand(x_i^t), \quad C = k \times \epsilon_{ps}. \tag{3}$$

Thus, at every single complete epoch E_k, the training examples are randomly sampled ϵ_{ps} times, and during the whole training process the examples are randomly sampled $C = K \times \epsilon_{ps}$ times. This novel strategy is used for training CNN networks for face anti-spoofing application. This training strategy is named as $B_sRS - \epsilon_{ps}Sec - 1E(B_sRandom\ Samples - \epsilon_{ps}sub - epochs\ Count - 1Epoch)$,which is used here to train a VGG-11 network and its derived networks. For each forward-pass through the CNN network, 25 face images are randomly sampled from the training dataset. Each forward-pass through a CNN network correspond to a single sub-epoch, and 60 sub-epochs count for 1 complete epoch. Fig. 2 shows training of a VGG-11 network using the conventional and the proposed training approach respectively. It can be clearly seen in Fig. 2 (a), (b) that using the conventional training method, training loss reduces in a 25 epochs, however with the proposed training method, the VGG-11 network can be easily trained for 500 epochs without any over-fitting as shown in Fig. 2(c).

Conventionally, training a deep CNN network with small scale data lead to over-fitting. This occur mainly due to the large capacity of deep CNN networks as compared to the training data. Adding a dropout can circumvent this problem to certain extent by providing regularization, however since the face anti-spoofing dataset is

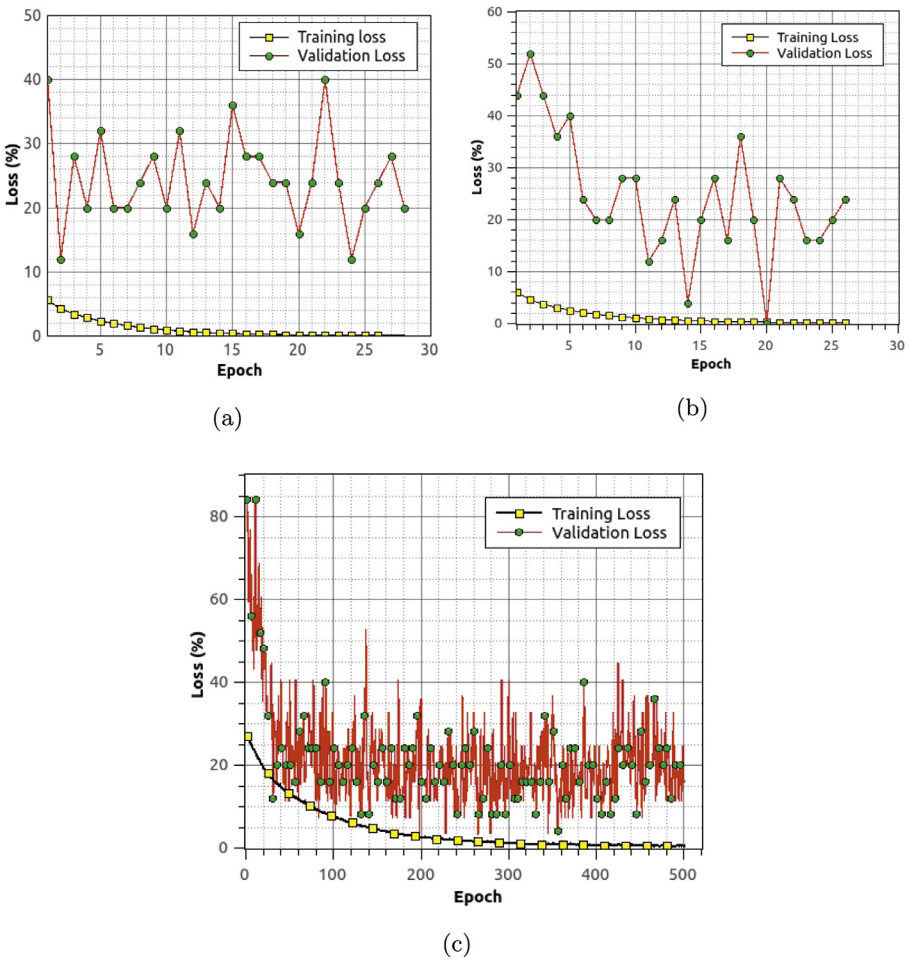

Fig. 2. Training a VGG-11 network using conventional training method and proposed strategy for low-sample size data. (a) Training loss and validation loss for a standard VGG-11 network using conventional training technique. (b) Training loss and validation loss for a modified VGG-11 using conventional training technique. (c) Training loss and validation loss for a modified VGG-11 network using proposed training technique.

small, over-fitting still occurs. Further, in conventional approach for training CNN networks, data as a whole was randomized once and passed through the CNN networks. However, with this approach the deep CNN networks with large capacity gains the knowledge of the whole training data in few epochs, which reduces their learning ability quickly as evident from Fig. 2(a), (b). However, with proposed method, we trained the network without over-fitting for 500 epochs as shown in Fig. 2(c). Although, in the proposed approach the amount of over-fitting is reduced substantially, there is a likelihood that some of the samples in the training dataset may not pass through the CNN network during the complete training stage. However, since the face anti-spoofing databases have limited samples, the probability of training-samples not being included during training stage is very low. The main attributes of reducing over-fitting in the proposed approach are as follows:

1. First, the network can be trained on more than a single video-frame as compared to the previous approaches used for face liveness detection (Yang et al., 2014). That said, the network is then able to learn more about the discriminative features in the input data.
2. Second, the CNN network can be trained from scratch using end-to-end learning for face anti-spoofing systems and applications as compared to other approaches that used CNN network using transfer learning or additional classifiers for face anti-spoofing (Li et al., 2016; Menotti et al., 2015; Yang et al., 2014).
3. Third, training on a known indexed single frame from the input video-data can make the network deterministic regarding the input data and may lead to over-fitting. Thus, providing more frames from the video-data can circumvent the problem of over-fitting and can improve the generalization capabilities of the CNN network. More importantly at each single epoch E_k, the training examples are randomly sampled ϵ_{ps} times, which means that at the end of a complete epoch the sampling process is repeated $C = K \times \epsilon_{ps}$ times. That mean that a batch of training examples presented to the CNN network at each single epoch is not deterministic but rather random. Thus, the introduction of these random batches of training data leads to innovation in the CNN network's cost-function at every sub-epoch and the CNN network weights are dynamically updated at each complete epoch. This can also be considered as a dropout mechanism in the input data as was done similar for CNN network regularization (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). Thus, for example, some of the training samples may not be given as an input to the CNN network during a single complete epoch but is introduced to the CNN network after some epochs. Thus, the introduction of these new samples results in innovation in the CNN's cost-function and hence can alleviate the problem of over-fitting to a greater extent.
4. Fourth, the network training time is reduced substantially. Table 1 shows a comparison between training time required

Table 1
Comparison between total time taken for training CNN network using conventional training approach and the proposed training approach.

	Conventional CNN training technique	Proposed CNN training technique
Epochs	26	500
Time (hours)	6.74	5.5
Time saved %	$\frac{6.74-5.5}{6.74} \times 100 = 18.39\%$	

Table 2
VGG-11 and its derived networks.

Network A	Network B	Network C	Network D
		$96 \times 96 \times 3$	
3×3, 64R	7×7, 64R	7×7, 64R	3×3, 64R
2×2 mp	2×2 mp	2×2 mp, Dp 0.5	2×2 mp, Dp 0.5
3×3, 128R	5×5, 128R	5×5, 128R	3×3, 128R
2×2 mp	2×2 mp	2×2 mp, Dp 0.5	2×2 mp, Dp 0.5
3×3, 256R	3×3, 256R	3×3, 256R	3×3, 256R
3×3, 256R	3×3, 256R	3×3, 256R	3×3, 256R
2×2 mp	2×2 mp	2×2 mp, Dp 0.5	2×2 mp, Dp 0.5
3×3, 512R	3×3, 512R	3×3, 512R	3×3, 512R
3×3, 512R	3×3, 512R	3×3, 512R	3×3, 512R
2×2 mp	2×2 mp	2×2 mp, Dp 0.5	$\times 2$ mp, Dp 0.5
3×3, 512R	3×3, 512R	3×3, 512R	3×3, 512R
3×3, 512R	3×3, 512R	3×3, 512R	3×3, 512R
2×2 mp	2×2 mp	2×2 mp, Dp 0.5	2×2 mp, Dp 0.5
		FC-4096R	
		FC-4096R	
		FC-4	
		Soft-max	

*R = RELU, mp = max-pool, Dp = Dropout, FC = Fully-Connected

for CNN to be trained for a fixed number of epochs. Since in the proposed training scheme, the batch-size is small and the sub-epochs are 60 which counts for 1 complete forward pass, the training time is reduced to 40 seconds per forward pass as compared to the conventional approach which takes almost 6.74 hours for only 26 epochs (forward pass).

The architecture of standard VGG-11 and its modified versions are shown in Table 2. For all CNN networks, the learning rate has been initially set to 0.01 for the first 100 epochs. The learning rate is then reduced by a factor of 0.1 for the next 100, 200 and 50 epochs respectively. Overall, the learning rate is reduced three times. The weight decay has been set to 0.005 with a dropout of 0.5 in the fully-connected layers for Network A, B, C and D respectively, and an additional dropout of 0.5 after every max-pooling layer in Network C and Network D. Rectified Linear Unit(ReLU) has been used as an activation function for all the networks. The video-frames are resized to 96×96 patch keeping the center part of the face patch and respecting the aspect-ratio. This is done by considering the presently adopted computer system limitations and GPU capabilities.

3.2. Evaluation metrics

Let H denote a set of human subjects, and x_i denotes each individual human subject, where i is an integer. Each subject also contains a set Z of image samples, which contains a subset of G genuine images and a subset of S spoofed images. Additionally, there is further categorization of a set S depending upon the type of spoofing attack represented by its samples. Denote the images in set S by A. Since, any s_m image presentation to a biometric system is generally considered as an attack, denoted by a, on the corresponding system, s_m can generally belong to n types attack. Thus, mathematically the set H, Z and A can be written by using (Eqs. (4)–(6)).

$$H = \{(G,S)|(G,S) \in Z \text{ and } Z \in x_i\}, \quad i \in \mathbf{Z}_+, \tag{4}$$

$$Z = \{z_1, z_2, \ldots, z_n | z_j \in G \text{ or } z_j \in S\}, \quad \{n, j\} \in \mathbf{Z}_+, \tag{5}$$

$$A = \{s_m | s_m \in A \text{ and } s_m \in \{a_1, \ldots, a_n\}\}, \{m, n\} \in \mathbf{Z}_+. \tag{6}$$

A CNN network for face anti-spoofing application has a predefined task of classifying an input image sample z_l of x_i user as either genuine g or spoofed s. A CNN network achieve this task by mapping the input raw pixels of an image sample z_l to the N-dimensional output probabilities vector $\mathbf{y} = \{y_1, y_2, \ldots, y_n\}$. The probabilities are then compared with a decision threshold value τ to determine the class (cls)of an image sample z_l. The output of a j_{th} CNN network with soft-max $\boldsymbol{\sigma}$ at its output can be defined by using (Eqs. (7) and (8)).

$$\mathbf{y_{l,j}} = \sigma(CNN_j(z_l)), \quad (l, j) \in \mathbf{Z}_+, \tag{7}$$

$$\sigma(CNN_j(z_l)) = \frac{e^{CNN_j(z_l)}}{\sum_N e^{CNN_j(z_l)}}. \tag{8}$$

For binary face classification problem or face liveness detection $\mathbf{y} = \{y_1, y_2\}$. Where, y_1 represents the probability of true class. Thus, for binary classification, the cls of an image sample z_l can be determined by considering only y_1 . The decision threshold can then be written by using (Eq. (9)).

$$cls(z_l \in G, S | y_{l,j}) = \begin{cases} z_l \in G, & \text{if } y_1 > \tau \\ z_l \in S, & \text{Otherwise} \end{cases} \tag{9}$$

The soft-max function at the output of a CNN network gives a normalized output, which can be compared with a set of threshold values to get a list of False Acceptance Rate (FAR) and False Rejection Rate (FRR) that can then determine the overall system Equal Error Rate (EER), Half Total Error Rate (HTER) and Receiver Operating Point (ROC) curve.

The FAR is a probability of a j_{th} CNN network in accepting input image samples z_l, having ground truth label s_g (spoofed face), of a user x_i as genuine attempts g_l (live face). Whereas the FRR is a probability of j_{th} CNN network in rejecting input image samples z_l, having ground truth label g_g, of a user x_i as spoofing s_l attempts. Mathematically, they can be defined by using (Eqs. (10) and (11)) where the sum is over all K samples in the test set.

$$FAR(CNN_j) = \frac{\sum_{l=1}^{K} z_l^{x_i} = g_l^{x_i}}{Z}, \qquad z_l^{x_i} = s_g \tag{10}$$

$$FRR(CNN_j) = \frac{\sum_{l=1}^{K} z_l^{x_i} = s_l^{x_i}}{Z}, \qquad z_l^{x_i} = g_g \tag{11}$$

The EER is normally threshold independent, and determine the equilibrium point between the FAR and FRR ratio. However, in real-time systems it is very difficult to obtain an equilibrium point. Thus, EER is normally determined by using Receiving Operating Point (ROC) curve, which is the distribution of FRR against FAR at various threshold values (Toh, Kim, & Lee, 2008).

HTER is usually threshold dependent. Normally, for calculating HTER, the face anti-spoofing dataset has a split between training set O , testing set P and validation or development set D . The training set is used to train the face anti-spoofing system; the validation or development set is used to find the threshold value τ at which the FAR and FRR become equal and the performance of

the system is determined by computing the HTER on the test set at a threshold value determined by development set. This can be mathematically defined by using (Eq. (12))

$$HTER = \frac{FAR(\tau, P) + FRR(\tau, P)}{2} \tag{12}$$

3.3. Face liveness detection

In face liveness detection, the aim of CNN network is to classify an input face image sample z_l as either G class or S class. Since, state-of-the-art face anti-spoofing databases (like CASIA-FASD and Replay-Attack) have a division of S samples among multiple classes as compared to samples belonging to G classes, an issue of sample imbalance often occurs. For example, CASIA-FASD database provide 3 video samples for G class and 9 video samples for S class for each human subject. Thus, a G class for every subject in CASIA-FASD has only 25% of the total video samples. A system then trained for face liveness detection on such data, although gives high accuracy, could provide poor anti-spoofing capabilities and poor generalization of the system in cross-database analysis. To determine whether a face anti-spoofing system perform better in face liveness detection, HTER and EER values have been used. While an EER can provide a faithful determination of system efficiency in liveness detection; an HTER may result in poor system performance in cross-database analysis. This is because for an HTER, a fixed threshold value τ determined by development set, is used throughout the testing. Thus, for a dataset with an imbalance of samples between the G class and S class, an HTER value may give a wrong point than equilibrium point (i.e. the point where the difference between FAR and FRR is minimum) in test dataset, and thus results in a system either more prone to accepting fake image samples or rejecting genuine image samples respectively. Consider for example the case of using HTER value on the test set, a set threshold value (determined by development set) gives FRR = 0%, and FAR = 25%, and liveness detection accuracy of the system is 75%. Thus, HTER = 12.5%, however the system has a higher tendency of accepting 25% fake image samples. Thus, in the test sets the HTER might or might not point to the actual equilibrium point. Thus, in this paper, we train our proposed CNN networks by maintaining equal distribution of each type of image samples. Further to find a quantitative measure, independent of threshold value , a top-1 accuracy is introduced. In top-1 accuracy, the class having maximum probability is considered as the true class. The top-1 accuracy is a common approach adopted for object classification in CNN networks (Szegedy et al., 2015). Mathematically, top-1 accuracy can be defined by using (13).

$$cls(\mathbf{y}_{l,j}) = argmax_{idx}\{\mathbf{y}_{l,j}\} \tag{13}$$

The top-1 accuracy generally determines the point where the ROC curve intersect the EER line.

3.4. System design

The proposed system is combination of cascade systems consisting of face detector and CNN network. The block diagram of the system is shown in Fig. 1(b). For detecting the face regions Viola-Jones face detector (Viola & Jones, 2004) is used. The detected face regions are then fed to CNN network for determining the class of the input face image. Before training a CNN architecture for face anti-spoofing on a given dataset, the face images in the dataset are pre-processed, which usually includes mean centering and normalization. Since, state-of-the-art face anti-spoofing databases (like CASIA-FASD), mainly consists of video-data rather than single images, the proposed data-preparation approach is different from conventional CNN data-preparation approaches for face anti-spoofing. In conventional CNN based approaches for face anti-spoofing, only a limited portion of video-data, restricted to 10 to 20 frames of each video, were utilized as an input data to the CNN architecture. On contrary, we randomly select 100 frames from each video in the database and stored them in a disk. At training time, we detect the face area using Voila-Jones cascade classifier in the mini-batch image frames and normalize it before giving as an input to a CNN network.

4. Experiments and evaluation

For the experiments, first, all the networks are individually trained on each database for intra-database evaluation and cross-database evaluation. Second, the predictions from all the networks for intra-database and cross-database are used to calculate the evaluation metrics, i.e. Accuracy (Acc), EER, HTER and ROC curve. Further, for each intra-database and cross-database evaluation respectively, the standard given protocols were followed for further tests as defined in Zhang et al. (2012), Chingovska et al. (2012) and Feng, Po, Li and Yuan (2016). The most important aspect in the present study is to assess the generalization capabilities of the proposed CNN compared to other state-of-the-art approaches for face liveness detection, i.e. to assess, whether the proposed CNN networks perform effectively on unseen face image samples for face liveness detection and spoofing attack classification. Thus for a fair comparison, we also provide results obtained by training CNN network using conventional approach and the proposed approach. In the following paragraphs, the details of the experimental setup, databases and protocols used for face liveness detection and spoofing attack classification are presented.

4.1. Databases used for experiments and evaluation

Two face anti-spoofing databases namely CASIA-FASD and Replay-Attack are used for the extensive evaluation of the proposed CNN networks on face liveness detection. It is worth noting that although many face anti-spoofing databases are available in literature for testing the performance of face anti-spoofing systems, however the adopted face anti-spoofing databases in this paper are more challenging as compared to other face anti-spoofing databases. The details of the database adopted in this paper are as follows.

4.1.1. CASIA-FASD

The CASIA-FASD database (from here on as CASIA) has a total of 50 human subjects. The database has been split into two sets: training set with 30 subjects and testing set with 20 subjects. For each subject, it provides 12 videos of which 3 are real access videos and 9 are fake access videos. For each subject, three imaging qualities are given, i.e. Low resolution(L), Normal resolution (N) and High resolution (H), and three presentation attacks are given, i.e. Wrapped photo attack (Wrapped), Cut photo attack (Cut) and Video tablet attack (Video). Three protocols namely Quality Test (QT), Fake Face Test (FFT) and Overall Test (OT) are given to evaluate the performance of face liveness detection system.

4.1.2. Replay-Attack

This database also contains 50 human subjects. The database has been split into three sets: train set, development set and test set. For the training set and development set, the database provides 360 non-overlapping video samples and for the test set, the database provides 480 non-overlapping video samples. For each subject in the database, three presentation attacks have been given, i.e. Print attack (Print), Mobile attack (Mobile) and High definition

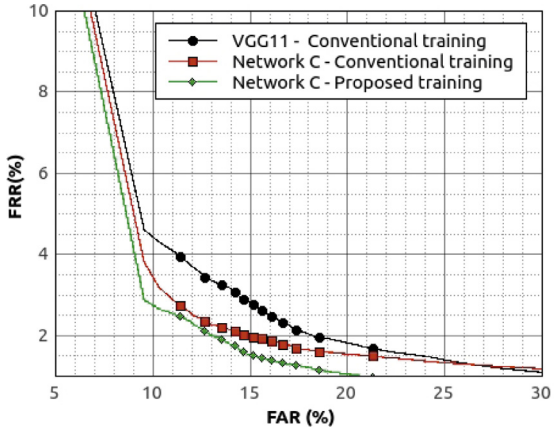

Fig. 3. ROC curve for training standard VGG-11 network using the conventional training technique, and training Network C using conventional and proposed training technique.

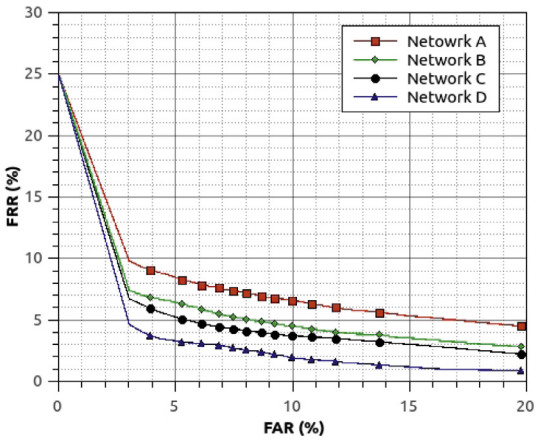

Fig. 4. ROC curve for intra-database test on CASIA.

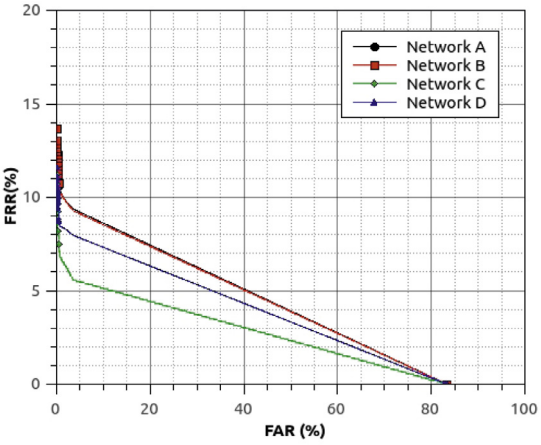

Fig. 5. ROC curve for intra-database on Replay-Attack.

attack (Highdef). Two lightning conditions i.e. adverse (A) and controlled (C), and two support conditions i.e. hand-based and fixed have been defined.

4.2. Liveness detection

Face liveness detection is binary classification problem. For face liveness detection, the elements of prediction vectors from each CNN network for intra-database and cross-database tests have been split into two categories, i.e. real access and fake access. Following this division, the proposed CNN networks are tested on the test protocols described previously for each database. In the following paragraph, a detail evaluation of the proposed CNN networks for face liveness detection on intra-database and cross-database is presented. Further, the proposed CNN networks are compared with state-of-the-art-face liveness detection algorithms on both intra-database and cross-database tests.

4.2.1. Intra-database evaluation

For intra-database evaluation, each network is individually trained on CASIA and Replay-Attack database respectively. Fig. 3 shows the ROC curve obtained by training a standard VGG-11 network and Network C using conventional training technique, and further training Network C using proposed training technique on CASIA-FASD database respectively. As can be observed in the Fig. 3, a comparatively lower EER is attained when the modified VGG-11 network is trained using the proposed training technique as compared to the conventional training technique. Further, since the proposed training technique lower down the time required for training CNN networks, we give the results obtained by only using the proposed training technique on all the CNN networks from here on.

Fig. 4 shows the ROC curve for the intra-database evaluation on CASIA test database. As can be depicted in Fig. 4, Network D achieve a low EER as compared to the rest of the CNN networks. Network D is an overall regularized network with a 3×3 kernel as a parameter regularization, and additional regularization in the form of dropout after every max-pooling layer. Addition of dropout regularize the CNN networks and further strengthen the generalization capabilities of CNN networks which can be clearly seen from the ROC curve as shown in Fig. 4. Table 3 shows, the corresponding threshold EER obtained from ROC curve and binary HTER obtained using top-1 accuracy for the proposed CNN networks on CASIA database test set and overall set (train set + test set) respectively. As can be seen in Table 3, Network D achieves an overall threshold EER of 4.59% on test set and threshold EER of 3.34% on overall set respectively. The overall set is included in the evaluation because the proposed training strategy uses data-randomization and there is a high chance that the CNN network may not get trained on some training samples indicated by a slightly high EER in the Overall set.

Fig. 5, shows the ROC curve of the proposed CNN network for intra-database test on Replay-Attack database. It can be depicted in Fig. 5 that Network D achieve an overall low threshold HTER of 5.74% on the test set. Similarly, Table 4, shows the HTER of the proposed CNN network for intra-database test on Replay-Attack database development set and test set. The development set of Replay-Attack database is used for each CNN network to determine and set a threshold value τ for the Replay-Attack database test set. The slight increase of HTER on the overall set clearly signify that some of the samples were not utilized during the training phase because of data randomization, however the HTER is well within the acceptable range.

Table 5 shows the comparison of intra-database results with other state-of-the-art techniques that used method like CNN networks. As can be seen in Table 5, the proposed CNN architectures perform consistently in intra-database analysis as compared to other state-of-the-art approaches. From Table 5, it can be see that the final HTER for intra-database test on CASIA is slightly higher than the Li et al. (2016), Siddiqui et al. (2016) and Manjani et al. (2017) and similarly for Replay-Attack database the HTER is higher than the Menotti et al. (2015), Feng, Po, Li, Xu et al. (2016), Pinto, Pedrini, Schwartz, and Rocha (2015), Siddiqui et al. (2016) and Manjani et al. (2017). This indicate that

Table 3
Intra-database results: Classification accuracy in (%) and EER in (%) on CASIA-FASD database.

Protocol Test	Network A		Network B		Network C		Network D	
	EER threshold	HTER binary	EER threshold	HTER binary	EER threshold	HTER binary	EER threshold	HTER binary
Test Set	7.37	6.12	6.88	6.92	5.09	4.61	4.59	4.81
Overall set	3.34	3.37	4.96	4.86	3.56	3.06	3.34	3.37

Table 4
Intra-databse results: Classification accuracy in (%) and EER in (%) on Replay-Attack database.

Protocol Test	Network A		Network B		Network C		Network D	
	HTER threshold	HTER binary	HTER threshold	HTER binary	HTER threshold	HTER binary	HTER Actual	HTER binary
Test Set	6.44	5.71	6.94	6.21	8.81	6.93	5.74	5.33
Development Set*	7.52	6.05	8.95	6.60	8.41	6.21	7.68	6.32
Total	6.98	5.88	7.95	6.41	8.61	6.57	6.71	5.83

*EER = HTER

Table 5
Intra-database results: Comparison with other state-of-the-art method (Liveness Detection).

	Intra-database	
Method	CASIA (Test) HTER (%)	Replay-Attack (Test) HTER (%)
DPCNN (Li et al., 2016)	4.5	6.1
SpoofNet (Menotti et al., 2015)	–	0.75
LSTM + CNN (Xu et al., 2015)	–	5.93
Non-Linear Diffusion (Alotaibi & Mahmood, 2017)	–	10
Multi-cues Integration + NN (Feng, Po, Li, Xu et al., 2016)	5.83[a]	0
Pinto et al. (2015)	14.3	2.8
Siddiqui et al. (2016)	3.8	0
DDGL (Manjani et al., 2017)	1.3	0
LiveNet	4.59[a]	5.74

[a] EER = HTER

Table 6
Cross-database results: Liveness detection accuracy and HTER in % respectively. Training set: CASIA, Evaluation set: Replay-Attack.

	Network A		Network B		Network C		Network D	
	HTER threshold	HTER binary	HTER threshold	HTER binary	HTER threshold	HTER binary	HTER threshold	HTER binary
Training Set	11.78	12.81	13.92	15.41	8.33	16.30	18.53	19.17
Test Set	14.25	13.34	12.80	13.06	8.39	16.03	17.30	19.14
Development Set*	14.92	15.13	14.61	15.66	8.61	17.70	19.23	19.39

*EER = HTER

on one hand, continuous data-randomization can help to train deep CNN networks on small scale database without over-fitting; on the other hand, it may prevent some samples to pass through CNN network that results a slight increase in the HTER in intra-database tests.

It is further emphasizing here, that an algorithm that gives low HTER on one database might give higher HTER on another database. Thus, the generality of the system is an important aspect of the system design, particularly when the system is designed for real-time scenario. Thus, a cross-database analysis is also presented in the following section.

4.2.2. Cross-database evaluation

To check whether the proposed approach generalized well to the unknown face spoofing attacks, a cross-database evaluation is performed. It must be noted here, that although some approaches for face anti-spoofing application perform remarkably well in intra-database evaluation, however they have a lower accuracy in cross-database evaluation. Fig. 6 shows the ROC plot for cross-database evaluation on Replay-Attack database, i.e. the network trained on CASIA database is tested on Replay-Attack database. From Fig. 6, and Table 6 it can be clearly seen that all proposed CNN networks provide good generalization scores. It can be seen in Table 6, that an all time lower HTER of 8.39% is attained on Replay-Attack(test) set using CNN Network C.For cross-database evaluation on Replay-Attack database, the Replay-Attack development dataset is used for CNN network trained on CASIA training database to determine a threshold value τ. Then, HTER values for liveness detection on train sets and test sets of Replay-Attack database are calculated. Similar process is repeated for CASIA database, i.e. the CNN networks trained on Replay-Attack datasets are evaluated using the train set and test of CASIA database. Fig. 7 and Table 7 shows the corresponding threshold HTER and binary HTER values for cross-database evaluation on CASIA database. As can be seen in Table 7, a lower HTER of 19.12% is attained on CASIA (test)set using CNN Network C.

Table 8 shows a comparison of the proposed approach with other state-of-the-art approaches for cross-database evaluation. It can be clearly seen that the approaches having low EER or HTER in intra-database evaluation have higher EER or HTER in cross-database evaluation. The proposed CNN networks provide significantly lower HTER values in cross-database evaluation, i.e. 8.39% on Replay-Attack and 19.12% on CASIA, outperforming other state-of-the-art approaches in cross-database evaluation, which shows that the proposed CNN networks have better generalization ability as compared to other state-of-the-art approaches. Further, the CNN networks were trained by utilizing data-randomization in the face

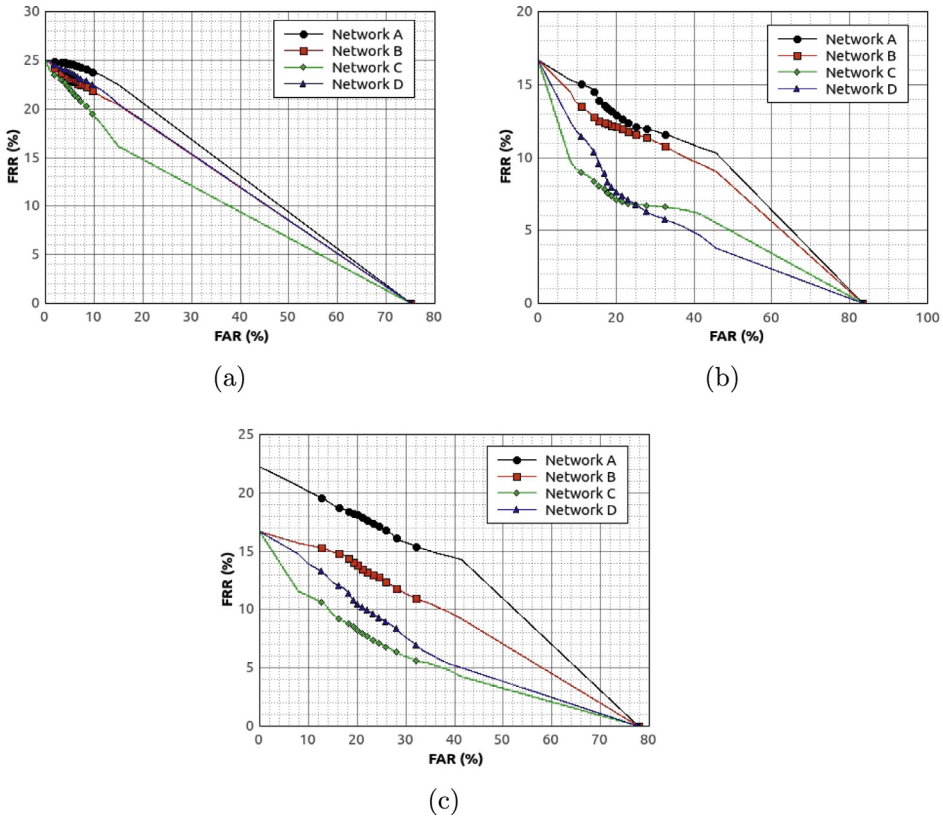

Fig. 6. ROC curve for cross-database test on Replay-Attack (a) Train data (b) Development data (c) Test data.

Table 7
Cross-databse results: Liveness detection accuracy and HTER in % respectively. Training set: Replay-Attack, Evaluation set: CASIA.

	Network A		Network B		Network C		Network D	
	EER threshold	HTER binary	EER threshold	HTER binary	EER threshold	HTER binary	EER threshold	HTER binary
Training Set	18.57	13.91	19.04	13.78	19.45	15.25	19.14	13.94
Test Set	18.83	13.24	20.20	13.82	19.12	14.96	21.84	13.55
Overall Set	18.73	13.51	19.74	13.80	19.25	15.08	20.76	13.70

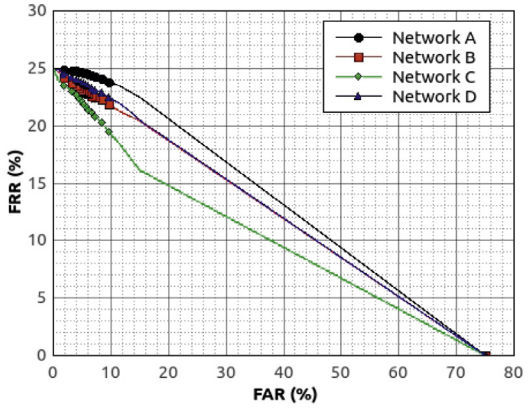

Fig. 7. ROC for cross-database test on CASIA.

Table 8
Cross-database Results: Comparison with other state-of-the-art method (Liveness Detection).

Method	Cross-database	
	CASIA (Test) HTER (%)	Replay-Attack (Test) HTER %
Pinto et al. (2015)	50	34.4
Siddiqui et al. (2016)	44.6	35.4
CNN (Yang et al., 2014)	38.11	23.78
DDGL (Manjani et al., 2017)	27.4	22.8
LiveNet	**19.12**	**8.39**

anti-spoofing datasets, i.e a different combination of face samples were fed to the CNN networks during each forward pass, that enable the network to learn more about intra-class variation thus improving the generalization and robustness of the learned features to unknown types of attacks.

In face anti-spoofing databases, train and test data respectively have been collected under the same conditions (illumination, temperature, head pose). Therefore the remarkable performance of classifiers on intra-database face liveness detection tests is evident. However, the same algorithms struggle to correctly classify the face image as live face or fake face in cross-database tests. The reason for this degradation in the performance of other algorithms in cross-database face liveness detection tests is because, the test database is completely different from the training database with different conditions (illumination, head pose). However, as compared to other state-of-the-art approaches, our proposed strategy

significantly improve the performance of face liveness detection in cross-database tests by lowering the HTER by 8.28% on CASIA and 14.28% on Replay-Attack database respectively.

5. Discussion

The proposed method provides an effective way of leveraging the generalization capabilities of deep CNN networks for cross-database face anti-spoofing. The data-randomization (like bootstrapping) approach is an effective way of preventing CNN networks from over-fitting caused by small training data. However, in the proposed approach, we randomly sampled small batches from the whole training dataset in a naive fashion. Thus, there is a chance that some of the samples, having different properties from the other samples in the database, may not pass through the CNN network. This is evident by a slight increase in the HTER in intra-database tests. However, in cross-database tests the HTER is significantly lower than the other state-of-the-art approaches. The proposed data-randomization method can be attributed to dropout mechanism used for regularizing the CNN networks. Since the data is randomly sampled, some of the training data is prevented to pass through the CNN network. After few epochs, some samples of the data that is prevented is passed through the CNN network. With the introduction of this new data to the CNN network, the loss is increased that results in weight adjustment of the hidden layers and hence the CNN network learning continuous without over-fitting on the training data.

Further, the training time is reduced substantially, by using small random batches, for training CNN network on small scale database. On contrary, training using the proposed technique on large scale database with small random batches with high-end GPU may result in an increase in the overall training time. Therefore, while using the proposed method for large scale data, the batch-size need to be set in according to the size the training data. For a reference, in this work the CASIA database has 24,000 frames while the Replay-Attack database has 36,000 frames for Development set, 36,000 for training set and 48,000 frames for test set.

6. Conclusion and future work

This paper proposed an efficient approach for face liveness detection when the training data is limited. The proposed approach utilizes continuous data-randomization in the form of small mini-batches during training a deep CNN network. The proposed approach is reliable in both intra-database and cross-database face liveness detection problems. Particularly for cross-database scenarios, the proposed approach significantly reduced the HTER. The proposed method attained an HTER of 19.12% for CASIA database and an HTER of 8.39% for Replay-Attack database in cross-database tests respectively. Further, the proposed strategy reduces the training time substantially by training the network in smaller and random batches. The data-randomization approach proposed in this work is similar to bootstrapping technique, which is quite effective in predicting the class of unknown samples using small scale population for training.

Future work include the extension of the proposed method to much deeper CNN networks like GoogleNet and ResNet and their performance evaluation on face anti-spoofing databases. Further, since the batch-size and sub-epochs are the hyper-parameters in the proposed work, we will further analyze the effect of varying batch-size and sub-epochs and vice versa on the generalization abilities of CNN networks. Additionally, inclusion of various schemes like data-augmentation, hybrid-CNN networks and extension of the proposed framework in detecting spoofing attacks in other medias are among the future work considered. Finally, since face-liveness detection is a binary classification problem, the inclusion of multi-class face anti-spoofing techniques will be considered in the future research that will further strengthen the abilities of counter measures for face anti-spoofing systems and applications.

Acknowledgment

The work described in this paper was substantially supported by a grant from the City University of Hong Kong, Kowloon, Hong Kong with Project number of 7004430.

References

Alotaibi, A., & Mahmood, A. (2017). Deep face liveness detection based on nonlinear diffusion using convolution neural network. *Signal, Image and Video Processing, 11*(4), 713–720.

Boulkenafet, Z., Akhtar, Z., Feng, X., & Hadid, A. (2017). Face anti-spoofing in biometric systems. In *Biometric security and privacy* (pp. 299–321). Springer.

Chingovska, I., Anjos, A., & Marcel, S. (2012). On the effectiveness of local binary patterns in face anti-spoofing. In *Biometrics special interest group (biosig), 2012 biosig-proceedings of the international conference of the* (pp. 1–7). IEEE.

Choudhury, B., Then, P., Issac, B., Raman, V., & Haldar, M. K. (2018). A survey on biometrics and cancelable biometrics systems. *International Journal of Image and Graphics, 18*(01), 1850006.

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

Feng, L., Po, L.-M., Li, Y., & Yuan, F. (2016). Face liveness detection using shearlet-based feature descriptors. *Journal of Electronic Imaging, 25*(4). 043014–043014

Galbally, J., Alonso-Fernandez, F., Fierrez, J., & Ortega-Garcia, J. (2012). A high performance fingerprint liveness detection method based on quality related features. *Future Generation Computer Systems, 28*(1), 311–321.

Galbally, J., Marcel, S., & Fierrez, J. (2014). Biometric antispoofing methods: A survey in face recognition. *IEEE Access, 2*, 1530–1552.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

Li, H., Lin, Z., Shen, X., Brandt, J., & Hua, G. (2015). A convolutional neural network cascade for face detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5325–5334).

Li, L., Feng, X., Boulkenafet, Z., Xia, Z., Li, M., & Hadid, A. (2016). An original face anti-spoofing approach using partial convolutional neural network. In *Image processing theory tools and applications (IPTA), 2016 6th international conference on* (pp. 1–6). IEEE.

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

Määttä, J., Hadid, A., & Pietikäinen, M. (2012). Face spoofing detection from single images using texture and local shape analysis. *IET Biometrics, 1*(1), 3–10.

Manjani, I., Tariyal, S., Vatsa, M., Singh, R., & Majumdar, A. (2017). Detecting silicone mask based presentation attack via deep dictionary learning. *IEEE Transactions on Information Forensics and Security*.

Marcel, S. (2013). Beat–biometrics evaluation and testing. *Biometric Technology Today, 2013*(1), 5–7.

Menotti, D., Chiachia, G., Pinto, A., Schwartz, W. R., Pedrini, H., Falcao, A. X., et al. (2015). Deep representations for iris, face, and fingerprint spoofing detection. *IEEE Transactions on Information Forensics and Security, 10*(4), 864–879.

Nguyen, K., Fookes, C., Jillela, R., Sridharan, S., & Ross, A. (2017). Long range iris recognition: A survey. *Pattern Recognition, 72*, 123–143.

Pinto, A., Pedrini, H., Schwartz, W. R., & Rocha, A. (2015). Face spoofing detection through visual codebooks of spectral temporal cubes. *IEEE Transactions on Image Processing, 24*(12), 4726–4740.

Ramachandra, R., & Busch, C. (2017). Presentation attack detection methods for face recognition systems: A comprehensive survey. *ACM Computing Surveys (CSUR), 50*(1), 8.

Sajjad, M., Khan, S., Hussain, T., Muhammad, K., Sangaiah, A. K., Castiglione, A., et al. (2018). Cnn-based anti-spoofing two-tier multi-factor authentication system. *Pattern Recognition Letters*.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815–823).

Siddiqui, T. A., Bharadwaj, S., Dhamecha, T. I., Agarwal, A., Vatsa, M., Singh, R., et al. (2016). Face anti-spoofing with multifeature videolet aggregation. In *Pattern recognition (ICPR), 2016 23rd international conference on* (pp. 1035–1040). IEEE.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15*(1), 1929–1958.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).

Toh, K.-A., Kim, J., & Lee, S. (2008). Biometric scores fusion based on total error rate minimization. *Pattern Recognition, 41*(3), 1066–1082.
Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision, 57*(2), 137–154.
Waris, M.-A., Zhang, H., Ahmad, I., Kiranyaz, S., & Gabbouj, M. (2013). Analysis of textural features for face biometric anti-spoofing. In *Signal processing conference (EUSIPCO), 2013 proceedings of the 21st European* (pp. 1–5). IEEE.
Xu, Z., Li, S., & Deng, W. (2015). Learning temporal features using lstm-cnn architecture for face anti-spoofing. In *Pattern recognition (acpr), 2015 3rd IAPR asian conference on* (pp. 141–145). IEEE.
Yang, J., Lei, Z., & Li, S. Z. (2014). Learn convolutional neural network for face anti-spoofing. arXiv preprint, arXiv:1408.5601.
Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D., & Li, S. Z. (2012). A face antispoofing database with diverse attacks. In *Biometrics (icb), 2012 5th IAPR international conference on* (pp. 26–31). IEEE.