Improving Cross-database Face Presentation Attack Detection via Adversarial Domain Adaptation

Guoqing Wang[1,3], Hu Han[*,1,2], Shiguang Shan[1,2,3,4], and Xilin Chen[1,3]

[1]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China
[2]Peng Cheng Laboratory, Shenzhen, China
[3]University of Chinese Academy of Sciences, Beijing 100049, China
[4]CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai, China
{guoqing.wang}@vipl.ict.ac.cn, {hanhu, sgshan, xlchen}@ict.ac.cn

Abstract

Face recognition (FR) is being widely used in many applications from access control to smartphone unlock. As a result, face presentation attack detection (PAD) has drawn increasing attentions to secure the FR systems. Traditional approaches for PAD mainly assume that training and testing scenarios are similar in imaging conditions (illumination, scene, camera sensor, etc.), and thus may lack good generalization capability into new application scenarios. In this work, we propose an end-to-end learning approach to improve PAD generalization capability by utilizing prior knowledge from source domain via adversarial domain adaptation. We first build a source domain PAD model optimized with triplet loss. Subsequently, we perform adversarial domain adaptation w.r.t. the target domain to learn a shared embedding space by both the source and target domain models, in which the discriminator cannot reliably predict whether a sample is from the source or target domain. Finally, PAD in the target domain is performed with k-nearest neighbors (k-NN) classifier in the embedding space. The proposed approach shows promising generalization capability in a number of public-domain face PAD databases.

1. Introduction

Biometric technologies such as FR are widely used in our daily life, e.g., in smartphone unlock, access control, and payment. It is well known that most of existing FR systems [28] are vulnerable to face presentation attacks (PA), e.g., a printed face on paper (print attack), replaying a face video on a screen (replay attack), wearing a face mask (3D mask attack), etc. Since an authorized user's face images can be easily obtained with a smartphone camera or from social media, which can be used for launching attacks against genuine users. Therefore, face PAD is an urgent problem to be solved. In recent years, a number of approaches have been proposed to handle print attack, replay attack, and 3D mask attack, respectively. Assuming that there are inherent disparities between live and spoof faces, many early PAD approaches utilized hand-crafted features for binary (live vs. spoof) classification with a

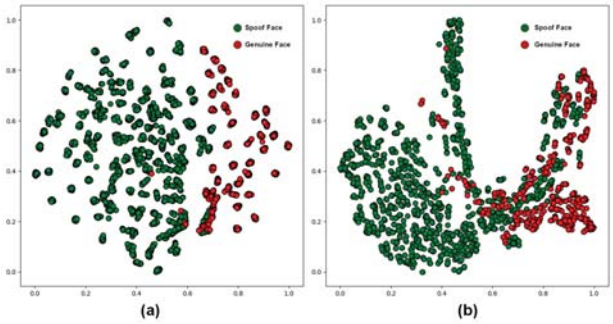

Figure 1. 2D visualization of the genuine and spoof face images from CASIA [35] and Idiap [6] with deeply learned features by ResNet-18 [12]. (a) The model trained on CASIA is tested (used for feature extraction) on CASIA (intra-database testing). (b) The model trained on CASIA is tested (used for feature extraction) on Idiap (cross-database testing). We observe that a model trained on the source domain does not generalize well to the target domain.

*Corresponding author.
978-1-7281-3640-0/19/$31.00

SVM model [5, 8, 15, 23, 31, 27, 26]. These methods have proven to be computationally efficient and work well under intra-database testing scenarios. However, the hand-crafted feature based methods did not show good generalization ability into a new application scenario [31]. With the success of deep learning, e.g., Convolutional Neural Networks (CNNs) [16] in many computer vision tasks, recent PAD approaches utilized CNNs for end-to-end face PAD or representation learning followed by binary classification using SVM [25, 33]. For example in [18], the deep learning feature based methods show improved performance than the traditional hand-crafted feature based methods under intra-database scenarios; however, they also found that the deep learning based methods may also not generalize well under cross-database testing scenarios (see a visualization in Fig. 1). The reason is that the differences between genuine and spoof faces may consist of multiple factors, such as skin detail loss, color distortion, moiré pattern, shape deformation, and spoof artifacts. The presence of these factors under two scenarios (databases) can be dramatically different; thus it is not enough to simply treat PAD as a common two-class classification problem. To improve the robustness of PAD, some scenario invariant auxiliary information such as depth and rPPG signals were also utilized to distinguish between live and spoof faces [2, 21]. Recently, domain adaptation (DA) has been utilized to mitigate the gap between the target domain and the source domain during face PAD [18, 29, 17].

In this paper, we focus on improving PAD generalization ability for cross-database PAD, and propose an end-to-end trainable PAD approach via unsupervised adversarial domain adaption (ADA). In particular, given the labeled genuine and spoof face images in source domain and unlabeled face images in the target domain, we aim to learn a joint embedding feature space for both the source domain and the target domain models in an adversarial way, while it is discriminative for distinguishing between the live and spoof face images in the source domain. Therefore, the proposed approach is able to leverage the prior knowledge from the source domain to perform more robust PAD in the target domain. Our approach is end-to-end trainable, and achieves promising results in cross-database face PAD on several public-domain databases (Idiap Replay-Attack (Idiap) [6], CASIA Face AntiSpoofing (CASIA) [35] and MSU-MFSD (MSU) [31]).

The main contributions of this work are three-fold: (i) a novel network architecture for improving cross-database PAD performance via adversarial domain adaptation (ADA) to leverage the prior knowledge from the source domain; (ii) utilizing metric learning in building PAD model in order to obtain more discriminative feature representation for live and spoof faces; and (iii) good generalization ability in both cross-database testing and intra-database testing scenarios.

2. Related Work

2.1. Face Presentation Attack Detection (PAD)

In the past few years, a large number of methods have been proposed for face presentation attack detection, which can be grouped into two categories: hand-crafted feature based methods and deep learning feature based methods.

1) Hand-crafted feature based methods: Since most face recognition (FR) systems are using commodity, print attack and replay attack become two major presentation attacks. The early works designed various hand-crafted features based on the observed discriminative texture cues between live and spoof faces, such as LBP [8, 23], LPQ [5], HoG [15], SIFT [26], SURF [5] and IDA features [31]. Some other works adopt hand-crafted features for face motion analysis such as eyes and mouth [24, 14] and 3D geometry analysis [20]. Given these different hand-crafted features classifiers such as SVM and LDA [3]. To reduce the influence of illumination variation and image conditions, some approaches convert the RGB images into HSV and YCbCr color space [4, 5] and Fourier spectrum space [19], and then extract the corresponding hand-crafted features.

The main advantages of these approaches are that they are usually computationally efficient and works well under intra-database testing scenarios. However, most of the times, it is not easy to collect the training data in advance, which has the same conditions as the testing scenarios.

2) Deep learning based methods: Heading into the era of deep learning, a large amount of research attempts to use CNN-based features or CNNs for face PAD [9, 25, 33] because of the significant performance improvement reported in many other computer vision tasks. CNN was used as a feature extractor for PAD, which is fine-tuned from ImageNet-pretrained CaffeNet and VGG-face [33]. Xu et al. [32] proposed to use CNN-LSTM to model multi-frame information. Liu et al. [21] observed the overfitting issue of softmax loss, and proposed a novelty framework based on auxiliary-driven loss to supervise the CNN learning process. Jourabloo et al. [13] inversely separated spoof noise from a spoof face, and then used it for spoof classification.

While the deep learning based methods show strong feature representation ability, and can be trained end-to-end, there are inherent constraints in fully leveraging the strong modeling capacity of deep models: (i) while a deep network usually requires a big training set, most pubic-domain face PAD datasets are small; (ii) face PAD datasets can be severely imbalanced because of medium or manners in launching presentation attack can be numerous; it is not possible to obtain training data for all these different types of presentation attacks.

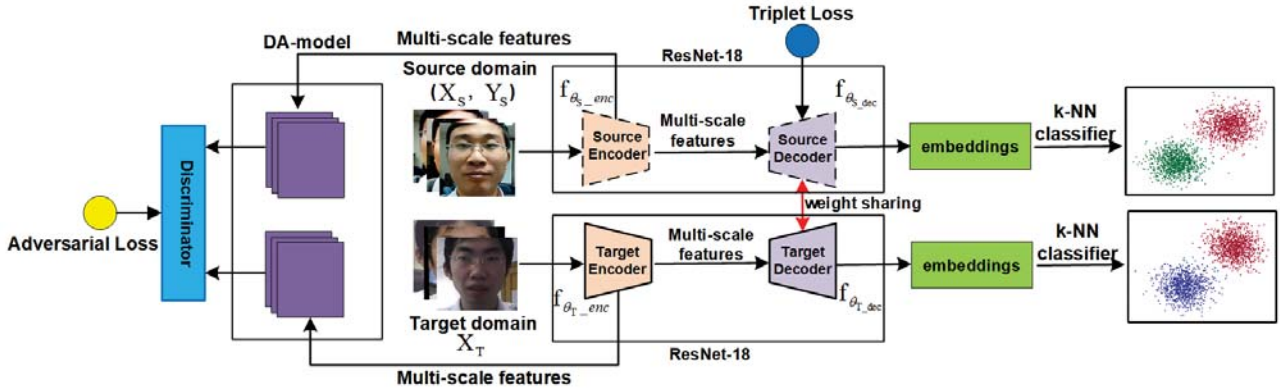

Figure 2. The overall diagram of the proposed approach for face presentation attack detection via adversarial domain adaptation. We first pre-train a source model optimized with triplet loss in source domain. Subsequently, we perform adversarial adaptation by learning a target model such that under the embedding space the discriminator cannot reliably predict whether a sample is from source domain or target domain. Finally, target images are mapped with the target model to the embedding space and classified with k-nearest neighbors classifier. Dashed lines indicate fixed network parameters.

2.2. Domain Adaptation for PAD

Domain adaptation aims at learning from a source domain a well performing model on a different but related target domain. Domain adaptation can be very useful when the training data in the target domain is very limited. There are several review papers for domain adaption [10, 22, 7]; here we only briefly summarize the approaches that utilize domain adaptation for addressing the face PAD. Yang et al. [34] proposed a person-specific face anti-spoofing approach based on a subject-specific domain adaptation method to synthesize virtual features, which assumes that the relationship between genuine and fake face images belonging to the same individual can be formulated as a linear transformation. Li et al. [18] proposed a novel framework utilizing unsupervised domain adaptation algorithms as Maximum Mean Discrepancy (MMD) for face anti-spoofing; the novelty of this framework lies in transferring the feature space of face samples from the labeled source domain to the unlabeled target domain. Recently, Generative Adversarial Networks (GANs) [11] have been applied to generate realistic images of objects including face images and have been extended in several fields. This has motivated us to design a novel adversarial domain adaptation method for face PAD.

3. Proposed Method

As shown in Fig. 2, our approach consists of two parts, i.e., source domain model learning and adversarial domain adaptation.

3.1. Source-domain Model Learning

Let (X_s, Y_s) denote the source domain face images X_s and the corresponding labels Y_s (live or spoof). Let X_t denote the target domain face images without known labels. Our source domain model aims to obtain a feature representation based on the labeled face images (X_s, Y_s)

$$\mathcal{F} = f_{\theta_S}(x), \mathcal{F} \in \mathbb{R}^d \tag{1}$$

which is informative for discriminating between live and spoof face images x. We use a deep neural network, i.e., ResNet-18 [12] to learn $f_{\theta_S}(\cdot)$. However, instead of using the commonly used cross-entropy loss, we choose to use triplet loss for metric learning to handle the big within-class diversity, particular for the spoof face class

$$\mathcal{L}(\theta_S) = \sum_{(x_i^a, x_i^p, x_i^n)} max(\|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^p)\|^2 - \|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^n)\|^2 + \alpha, 0) \tag{2}$$

Here, we want to ensure that an anchor image x_i^a of a specific class, no matter it is live or spoof, can be closer to an image of same class x_i^p (positive sample) than an image of different class x_i^n (negative sample). Optimizing Eq. (2) is to encourage that the features extracted from x_i^a to be closer to x_i^p than to x_i^n by at least a margin of α in the embedding space. Minimizing this term results in moving the anchor sample x_i^a towards positive samples x_i^p while keeping away from negative sample x_i^n under the embedding feature space. In addition, the live and spoof face images are expected to form two clusters under the embedding space, making it easy for discriminating between live and spoof classes using k-nearest neighbors (k-NN) classifier [30].

Triplet Selection. In order to ensure good network convergence and representation learning, it is crucial to find the hard triplets of samples. Given any x_i^a, we determine its

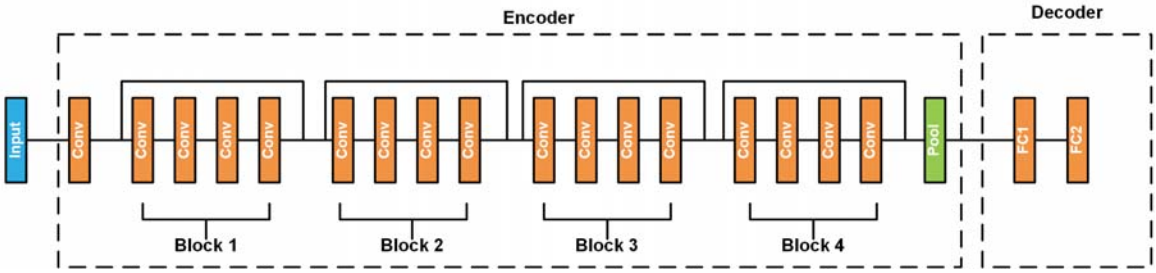

Figure 3. The network architecture of the domain adaptation part in the proposed presentation attack detection approach.

hard positive sample x_i^p by using the following rule:

$$\arg\max_{x_i^p} \|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^p)\|^2 \tag{3}$$

Similarly, we determine the hard negative sample x_i^n by using the following rule:

$$\arg\min_{x_i^n} \|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^n)\|^2 \tag{4}$$

Then, a triplet pair is composed of (x_i^a, x_i^p, x_i^n). Each triplet pair is used to compute the loss and update the parameters.

3.2. Adversarial Domain Adaptation

Since the face images X_t in the target domain do not have annotated labels, direct supervision (in terms of live vs. spoof labels) in the target domain is not possible. Therefore, we aim at building a PAD model $f_{\theta_T}(\cdot)$ that can leverage the source domain knowledge and effectively distinguish between live and spoof face images in the target domain. We propose an adversarial domain adaptation (ADA) to build the target domain PAD model $f_{\theta_T}(\cdot)$. In order to perform ADA, we divide both the source domain model and target domain model into two parts: encoder and decoder (see Fig. 2). The relationship of encoders, decoders, and the entire model f_θ are as follows:

$$\begin{aligned} f_{\theta_S}(x) &= f_{\theta_{S\_dec}}(f_{\theta_{S\_enc}}(x)) \\ f_{\theta_T}(x) &= f_{\theta_{T\_dec}}(f_{\theta_{T\_enc}}(x)) \end{aligned} \tag{5}$$

where the source encoder $f_{\theta_{S\_enc}}$ and target encoder $f_{\theta_{T\_enc}}$ are used to extract multi-scale features from the face images in source domain and target domain, respectively. The source decoder $f_{\theta_{S\_dec}}$ and target decoder $f_{\theta_{T\_dec}}$ map the multi-scale features extracted from encoders to embedding space $\mathcal{F}$. Our adversarial domain adaptation aims to obtain an embedding space $\mathbb{R}^d_{enc}$ which is shared by both the source domain encoder and the target domain encoder. Since this shared embedding space is discriminative for the live and spoof face images in the source domain, it is also expected to be discriminative for the unlabeled live and spoof face images in the target domain using the same k-NN classifier learned in Section 3.1. To achieve this, we simultaneously optimize $f_{\theta_{S\_enc}}$ and $f_{\theta_{T\_enc}}$ in the source and target domain.

Formally, let $\{\ell_1, ..., \ell_n\}$ be the individual layer index of a network (either f_{θ_S} or f_{θ_T}), and $f^\ell_{\theta_S}$ (or $f^\ell_{\theta_T}$) denotes the features of the l-th layer. The specific constraints is as follows:

$$\psi_{\ell_i}(f^{\ell_i}_{\theta_S}, f^{\ell_i}_{\theta_T}) = (f^{\ell_i}_{\theta_S} = f^{\ell_i}_{\theta_T})_{i\in\{1...n\}} \tag{6}$$

where n represents the number of layers in each model. We use this form of constraint to initialize the parameters of the target model. These equality constraints can easily be imposed within a convolutional network framework through weight sharing.

Adversarial Loss. Our adversarial domain adaptation aims at optimizing $f_{\theta_{S\_enc}}$ and $f_{\theta_{T\_enc}}$ to obtain a shared embedding space, such that under such a shared embedding space, the samples from the source domain are indistinguishable from the samples in the target domain. Therefore, we introduce a discriminator D, which aims to separate the source domain samples from the target domain samples in the embedding space. Here, D is optimized following a standard GAN loss, i.e.,

$$\begin{aligned} \mathcal{L}_D(\theta_{T_{enc}}, \theta_D) = &-\mathbb{E}_{x_s\sim X_s}[\log D_{\theta_D}(f_{\theta_{S\_enc}}(X_s))] \\ &-\mathbb{E}_{x_t\sim X_t}[\log(1 - D_{\theta_D}(f_{\theta_{T\_enc}}(X_t)))] \end{aligned} \tag{7}$$

where θ_{S_enc} is the parameters for the source encoder, θ_{T_enc} is the parameters for the target encoder, θ_D is the parameter of D.

We did not optimize the adversarial domain adaptation directly using the minimax loss. Instead, we split the objective into two independent objectives, one for optimizing domain encoder and the other for optimizing the discriminator. The loss $\mathcal{L}_D$ for discriminator D remains unchanged. The target encoder loss $\mathcal{L}_E$ is defined as

$$\mathcal{L}_E(\theta_{T\_enc}, \theta_D) = -\mathbb{E}_{x_t\sim X_t}[\log D_{\theta_D}(f_{\theta_{T\_enc}}(X_t))] \tag{8}$$

The learned network parameters in Section 3.1 remain unchanged during this adversarial domain adaptation process,

Method	C → I	C → M	I → C	I → M	M → C	M → I	Avg
ResNet w/o DA	43.3	14.0	45.4	35.3	37.8	11.5	31.2
Yang et al. [34]	49.2	18.1	39.6	36.7	49.6	49.6	40.5
Li et al. [18]	39.2	14.3	**26.3**	33.2	**10.1**	33.3	26.1
Proposed method	**17.5**	**9.3**	41.6	**30.5**	17.7	**5.1**	**20.3**

Table 1. Cross-database testing performance (HTER in %) of the proposed method and the state-of-the-art methods.

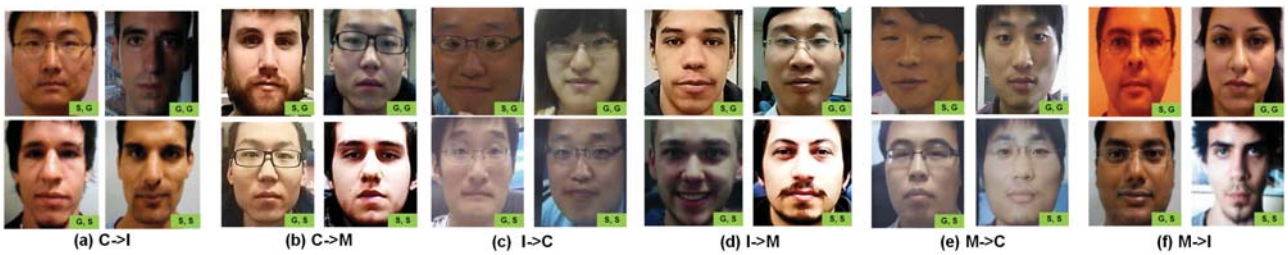

Figure 4. Examples of correct and incorrect PAD results by the proposed approach in six cross-database tests. The label "S, G" ("G, S") denote a spoof (genuine) face image is incorrectly classified as genuine (spoof) face image; "G, G" ("S, S") denote a genuine (spoof) face image is correctly classified as genuine (spoof).

and we only optimize the parameters for the target domain model.

In summary, the proposed adversarial domain adaptation is performed by optimizing the following three objective functions:

$$\begin{aligned}
\min_{f_{\theta_S}} \mathcal{L}(\theta_S) = \sum_{(x_i^a, x_i^p, x_i^n)} & max(\|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^n)\|^2 \\
& - \|f_{\theta_S}(x_i^a) - f_{\theta_S}(x_i^p)\|^2 + \alpha, 0) \\
\min_{D_{\theta_D}} \mathcal{L}_D(\theta_{T_enc}, \theta_D) = & -\mathbb{E}_{x_s \sim X_s}[\log D_{\theta_D}(f_{\theta_{S_enc}}(X_s))] \\
& -\mathbb{E}_{x_t \sim X_t}[\log(1 - D_{\theta_D}(f_{\theta_{T_enc}}(X_t)))] \\
\min_{f_{\theta_{T_enc}}} \mathcal{L}_E(\theta_{T_enc}, \theta_D) = & -\mathbb{E}_{x_t \sim X_t}[\log D_{\theta_D}(f_{\theta_{T_enc}}(X_t))]
\end{aligned} \tag{9}$$

4. Experimental Results

4.1. Databases

We evaluate our method on multiple databases to demonstrate its generalizability. We use the three public domain face databases for PAD including Idiap REPLAY-ATTACK [6], CASIA Face AntiSpoofing [35] and MSU-MFSD [31] for cross-database testing and intra-database testing.

4.2. Experimental Settings

We use an open source SeetaFace algorithm to do face detection and landmark localization. All the detected faces are then normalized to 256×256 based on 5 facial keypoints (two eye centers, nose, and two mouth corners).

https://github.com/seetaface/SeetaFaceEngine

We use ResNet-18 [12] as the backbone for both the source domain and target domain models. The network structure is shown in Fig. 3. There are 4 residual blocks in the encoder of each network and each block has 4 convolutional layers. The decoder of each network is a simple linear model containing 2 FC layers that aim to transform the encoded features into 128-D embedding feature vectors. The discriminator D in Fig. 2 consists of 3 fully connected layers: which have 256 hidden units, 128 hidden units, and two outputs nodes, respectively. Each of the first two layers uses a ReLU activation function.

We follow the state-of-the-art face PAD methods [31, 17, 18], and report Half Total Error Rate (HTER) [1] on the cross-database testing scenario. In intra-database testing, we use a five-fold cross-validation protocol [33, 18] for the CASIA and MSU datasets. For intra-database testing on Idiap, we follow its standard protocol.

4.3. Results

4.3.1 Cross-database Testing

We perform cross-database testing on CASIA, Idiap and MSU. We follow the same testing protocol as that in [18], i.e., train the model using all the images from dataset A, and test the model on a different dataset B (denoted as A→B). So, we have six tests in total: C→I, C→M, I→C, I→M, M→C, M→I, in which C, M, and I denote CASIA, MSU, and Idiap, respectively. Two state-of-the-art methods [34, 18] also used domain adaptation to improve the cross-database testing performance, and reported promising results. So we use both methods as our baseline algorithms. In addition, we also report the performance of our source

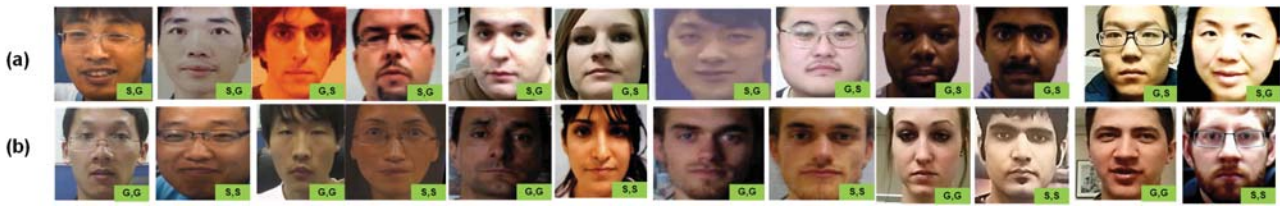

Figure 5. Examples of (a) correct and (b) incorrect PAD results by the proposed approach under intra-database tests on Idiap, CASIA, and MSU. The label "S, G" ("G, S") denote a spoof (genuine) face image is incorrectly classified as genuine (spoof) face image; "G, G" ("S, S") denote a genuine (spoof) face image is correctly classified as genuine (spoof).

Method	Testing Set			
	Idiap	MSU	CASIA	Avg
Resnet w/o DA	27.3	23.8	43.3	31.5
Yang et al. [34]	49.2	18.1	39.6	35.6
Li et al. [18]	33.3	14.3	**10.1**	19.2
Proposed method	**6.6**	**12.9**	37.8	**19.1**

Table 2. Cross-database testing performance (HTER in %) by individual methods, in which one of the three datasets is used for testing, and the other two datasets are used for training.

Method	Idiap HTER	CASIA EER	MSU EER
CoALBP (HSV) [18]	3.7	5.5	9.8
CoALBP (YCbCr) [18]	1.4	10.0	8.1
Deep learning [18]	2.1	7.6	**5.8**
Proposed	**1.4**	**3.2**	6.0

Table 3. Intra-database PAD performance (HTER and EER in %) on each of the three databases (Idiap, CASIA and MSU).

domain model (ResNet-18) that is directly tested without domain adaptation (ResNet w/o DA). The results are shown in Table 1.

We can observe that the two state-of-the-art face PAD methods using domain adaptation do not always achieve higher performance compared to the results by ResNet w/o DA. For example, under the tests of C→M and M→I, both baseline methods [34, 18] with DA performs worse than the baseline ResNet model without DA. Actually, DA in baseline method [34] does not lead to better performance than ResNet w/o DA. The possible reason is that the linear feature synthesis may work for the person identification scenarios in [34], but does not work well for the cross-database PAD tasks. The baseline method using DA [18] works better than ResNet w/o DA under most of the six tests. This suggests the usefulness of DA. The proposed approach with ADA performs better than the state-of-the-art face PAD method [18] under four of the six tests. This shows that the proposed ADA is more effective than the DA by [18] in mitigating the distribution gap between the source and target domains, while both DA methods are unsupervised. This suggests that the proposed approach has big potentiality when deploying a pre-trained face PAD model into unknown application scenarios.

Train dataset	Idiap HTER	CASIA EER	MSU EER
Single dataset	1.4	3.2	6.0
Multiple datasets	0.2	2.5	5.8

Table 4. Performance (HTER and EER in %) of intra-database testing with combine database.

We also notice that the proposed ADA may not lead to big performance improvement in some cases. For example, under the tests of I→C and I→M, the HTER after using ADA remains high, i.e., more than 40%. The possible reason is that the diversity of the spoof attack in Idiap is relatively limited compared with those in CASIA or MSU MFSD. By contrast, the diverse spoof attacks in either CASIA or MSU MFSD are helpful for identifying the spoof attacks in Idiap.

Fig. 4 shows some examples of correct and incorrect PAD results by the proposed approach under the six cross-dataset tests. We notice that most errors are caused when the testing face images have appearance variances such as over-saturated illumination, similar color distortions in both live and spoof face images, etc.

Considering that a single database remains two small in training deep face PAD models, we also provide evaluations when two of the three datasets are combined for training, and the third dataset is used for testing. The performance by the proposed approach and the baseline methods are shown in Table 2. We can see that leveraging bigger training set can further improve the cross-database testing performance. The proposed approach performs better than the state-of-the-art PAD method with DA [18], except on the CASIA dataset. The possible reason is that while the CASIA dataset mainly contains Asian people, the Idiap and MSU datasets have a relatively small number of Asian subjects. As a result, such a race distribution gap is not well handled during our ADA.

Method	C → I	C → M	I → C	I → M	M → C	M → I	Average
Proposed w/o ML&ADA	43.8	33.8	49.5	41.3	45.4	39.6	42.2
Proposed w/o ML	43.7	29.6	50.0	35.4	46.5	38.7	40.7
Proposed w/o ADA	43.3	14.0	45.4	35.3	37.8	11.5	31.2
Proposed (full method)	**17.5**	**9.3**	**41.6**	**30.5**	**17.7**	**5.1**	**20.3**

Table 5. Performance (HTER in %) of proposed method under ablation study in terms of metric learning (ML) and adversarial domain adaptation (ADA).

4.3.2 Intra-database Testing

Since many previous methods on face PAD reported their performance under an intra-database testing scenario, we also perform intra-database tests on CASIA, Idiap and MSU, respectively. In [18], CoALBAP features and deep learning features were reported to have the promising performance in intra-database testing. Therefore, we use the methods with CoALBP features in HSV and YCbCr color space and deep learning features [18] as our baselines. In particular, HTER is reported for the Idiap database, and EER is used for CASIA and MSU databases following [18].

The results are shown in Table 3, which shows that the proposed approach obtains much lower errors than the two state-of-the-art baseline methods in [18] on Idiap and CASIA, while very similar performance on MSU. The results by deep features based method in [18] and our method indicate the strong representation learning capacity of deep neural networks.

Fig. 5 shows some examples of correctly and incorrectly classified face images in intra-database tests on CASIA, Idiap and MSU. We notice that the incorrect face PAD results are mainly caused by the over-saturated illumination, color distortion, or bad image quality that diminishes the difference between genuine and spoof face images.

We also conduct the experiments by combining all the training data from three databases to learn a single model, and test it on the testing sets of each dataset. The results are shown in Table 4. As expected, combining multiple databases into a bigger dataset leads to better performance than training on each single database. This indicates that collecting a large PAD dataset in terms of the number of subjects and images remains important when the testing scenarios are known.

4.4. Ablation Study

We provide ablation study to validate the two components in the proposed face PAD method: (i) metric learning used in source model learning and (ii) adversarial domain adaptation (ADA). We study their influences by removing one component each time, and denote the corresponding model as 'Proposed w/o ML' and 'Proposed w/o ADA'. The results under cross-database testing are given in Table 5. We can see removing either component can lead to performance drop. This suggests that both components are useful in the proposed face PAD approach.

5. Conclusions

We propose an end-to-end approach to improve cross-domain face presentation attack detection (PAD) performance by utilizing the prior domain knowledge from source domain via adversarial domain adaptation. In the source PAD model learning, we use triplet loss for metric learning to handle the big within-class diversity. We then build the target domain PAD model by learning a joint embedding space for both the source and target domain through adversarial domain adaptation. The proposed approach outperforms the state-of-the-art face PAD methods under the challenging cross-database testing scenarios, and works well under intra-database testing scenarios. Our feature work includes utilizing the 3D face prior knowledge and physiological cues to improve the robustness of PAD. In addition, we will also study how to learn better representations that can minimize the influences by subject's identity, race, etc.

6. Acknowledgement

This research was supported in part by the Natural Science Foundation of China (grants 61732004, 61672496, and 61650202), External Cooperation Program of Chinese Academy of Sciences (CAS) (grant GJHZ1843), and Youth Innovation Promotion Association CAS (2018135).

References

[1] A. Anjos and S. Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. In *Proc. IJCB*, pages 1–7, 2011. 5

[2] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu. Face anti-spoofing using patch and depth-based CNNs. In *Proc. IJCB*, pages 319–328, 2017. 2

[3] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh. Computationally efficient face spoofing detection with motion magnification. In *Proc. CVPRW*, pages 105–110, 2013. 2

[4] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face anti-spoofing based on color texture analysis. In *Proc. ICIP*, pages 2636–2640, 2015. 2

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

[6] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Proc. BIOSIG*, 2012. 1, 2, 5

[7] G. Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint, arXiv:1702.05374*, 2017. 3

[8] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel. Lbp-top based countermeasure against face spoofing attacks. In *Proc. ACCV*, pages 121–132, 2012. 2

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

[10] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proc. ICCV*, pages 2960–2967, 2013. 3

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. NIPS*, pages 2672–2680, 2014. 3

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. CVPR*, pages 770–778, 2016. 1, 3, 5

[13] A. Jourabloo, Y. Liu, and X. Liu. Face de-spoofing: Anti-spoofing via noise modeling. *arXiv preprint, arXiv:1807.09968*, page 3, 2018. 2

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

[15] J. Komulainen, A. Hadid, and M. Pietikainen. Context based face anti-spoofing. In *Proc. BTAS*, pages 1–8, 2013. 2

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, pages 1097–1105, 2012. 2

[17] H. Li, P. He, S. Wang, A. Rocha, X. Jiang, and A. C. Kot. Learning generalized deep feature representation for face anti-spoofing. *IEEE Trans. Inf. Forensics Security*, 13(10):2639–2652, 2018. 2, 5

[18] H. Li, W. Li, H. Cao, S. Wang, F. Huang, and A. C. Kot. Unsupervised domain adaptation for face anti-spoofing. *IEEE Trans. Inf. Forensics Security*, 13(7):1794–1809, 2018. 2, 3, 5, 6, 7

[19] J. Li, Y. Wang, T. Tan, and A. K. Jain. Live face detection based on the analysis of fourier spectra. In *Proc. SPIE*, volume 5404, pages 296–304, 2004. 2

[20] S. Liu, P. C. Yuen, S. Zhang, and G. Zhao. 3D mask face anti-spoofing with remote photoplethysmography. In *Proc. ECCV*, pages 85–100, 2016. 2

[21] Y. Liu, A. Jourabloo, and X. Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *Proc. CVPR*, pages 389–398, 2018. 2

[22] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint, arXiv:1502.02791*, 2015. 3

[23] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Proc. IJCB*, pages 1–7, 2011. 2

[24] G. Pan, L. Sun, Z. Wu, and S. Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *Proc. ICCV*, 2007. 2

[25] K. Patel, H. Han, and A. K. Jain. Cross-database face anti-spoofing with robust feature representation. In *Proc. CCBR*, pages 611–619, 2016. 2

[26] K. Patel, H. Han, and A. K. Jain. Secure face unlock: Spoof detection on smartphones. *IEEE Trans. Inf. Forensics Security*, 11(10):2268–2283, 2016. 2

[27] K. Patel, H. Han, A. K. Jain, and G. Ott. Live face video vs. spoof face video: Use of moiré patterns to detect replay video attacks. In *Proc. ICB*, pages 98–105, 2015. 2

[28] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proc. CVPR*, pages 234–778, 2004. 1

[29] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proc. CVPR*, page 4, 2017. 2

[30] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Proc. NIPS*, pages 1473–1480, 2006. 3

[31] D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE Trans. Inf. Forensics Security*, 10(4):746–761, 2015. 2, 5

[32] Z. Xu, S. Li, and W. Deng. Learning temporal features using LSTM-CNN architecture for face anti-spoofing. In *Proc. ACPR*, pages 141–145, 2015. 2

[33] J. Yang, Z. Lei, and S. Z. Li. Learn convolutional neural network for face anti-spoofing. *arXiv preprint, arXiv:1408.5601*, 2014. 2, 5

[34] J. Yang, Z. Lei, D. Yi, and S. Z. Li. Person-specific face antispoofing with subject domain adaptation. *IEEE Trans. Inf. Forensics Security*, 10(4):797–809, 2015. 3, 5, 6

[35] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *Proc. ICB*, pages 26–31, 2012. 1, 2, 5