

Deep face recognition with clustering based domain adaptation

Mei Wang, Weihong Deng*

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

ARTICLE INFO

Article history:

Received 30 August 2019

Revised 18 December 2019

Accepted 2 February 2020

Available online 6 February 2020

Communicated by Dr. Qingshan Liu

Keywords:

Face recognition

Unsupervised domain adaptation

Pseudo-label

Face clustering

ABSTRACT

Despite great progress in face recognition tasks achieved by deep convolution neural networks (CNNs), these models often face challenges in real world tasks where training images gathered from Internet are different from test images because of different lighting condition, pose and image quality. These factors increase domain discrepancy between training (source domain) and testing (target domain) database and make the learnt models degenerate in application. Meanwhile, due to lack of labeled target data, directly fine-tuning the pre-learned models becomes intractable and impractical. In this paper, we propose a new clustering-based domain adaptation method designed for face recognition task in which the source and target domain do not share any classes. Our method effectively learns the discriminative target feature by aligning the feature domain globally, and, at the meantime, distinguishing the target clusters locally. Specifically, it first learns a more reliable representation for clustering by minimizing global domain discrepancy to reduce domain gaps, and then applies simplified spectral clustering method to generate pseudo-labels in the domain-invariant feature space, and finally learns discriminative target representation. Comprehensive experiments on widely-used GBU, IJB-A/B/C and RFW databases clearly demonstrate the effectiveness of our newly proposed approach. State-of-the-art performance of GBU data set is achieved by only unsupervised adaptation from the target training data.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

Benefiting from convolutional neural networks (CNNs) [1–5], deep face recognition (FR) has been the most efficient biometric technique for identity authentication and has been widely used in enormous areas such as military, finance, public security as well as our daily life. However, deep networks which perform perfectly on benchmark datasets may fail badly on real world applications. This is because the set of real world images is infinitely large and so it is hard for any dataset, no matter how big, to be representative of the complexity of the real world. One persuasive evidence is presented by P.J. Phillips' study [6] which conducted a cross benchmark assessment of VGG model [7] for face recognition. The VGG model, trained on over 2.6 million face images of celebrities from the Web, is a typical FR systems and achieves 98.95% on LFW [8] and 97.30% on YTF [9]. However, It only obtains 26%, 52% and 85% on Ugly, Bad and Good partition of GBU database, even if all of images in GBU are nominally frontal.

The main reason is a different distribution between training data (source domain) and testing data (target domain), referred to as domain or covariate shift. Visual examples of this domain

shift are shown in Fig. 1. Each dataset in Fig. 1 displays a unique “signature” and thus one can easily distinguish them only by these signatures, which proves the existence of significant discrepancies. The images in CASIA-WebFace [10] are collected from Internet under unconstrained environment and most of the figures are celebrities and public taken in ambient lighting; The GBU [11] contains still frontal facial images and is taken outdoors or indoors in atriums and hallways with digital camera; IJB-A [12] covers large pose variations and contains many blurry video frames. Sometimes, the images of GBU and IJB-A datasets may be closer to the ones in real life which are taken with digital camera under different shooting environments and contain larger variations.

To alleviate the problems caused by domain shift, the most popular approach is to fine-tune a pre-trained deep network's parameter on testing scenario with the supervision of data label. This straightforward strategy turns out to be problematic because it can be expensive or even infeasible to obtain required amount of labeled data in all possible testing scenarios. Moreover, more and more concerns on privacy may make the collection and human-annotation of the application-collected data become illegal in the future. Fortunately, unsupervised domain adaptation (UDA) is a promising technique aiming to address this problem, which learns a good predictive model for the target (testing) domain using labeled examples from the source (training) domain but only unlabeled examples from the target domain. Recently, many deep

* Corresponding author.

E-mail address: whdeng@bupt.edu.cn (W. Deng).



Fig. 1. Several sample images of three face databases. From left to right: (a) CASIA-Webface [10], (b) GBU [11] and (c) IJB-A [12]. Compared with CASIA-Webface, GBU is taken outdoors or indoors in atriums and hallways with different lighting conditions; IJB-A covers large pose variations and contains many blurry video frames.

UDA methods [13] try to learn more transferable representations through mapping both domains into a domain-invariant feature space, and then directly apply the classifier learned from only source labels to target domain, which produce boosted accuracy in various object recognition tasks [14–28].

In non-deep era, UDA was used for face recognition [29,30] in which the distributions of the two datasets are matched by learning a common shared space. In deep era, there have been many well-established deep UDA methods [13] for object classification and other computer vision applications. However, most of these methods are not applicable for the face recognition task at all. In particular, face recognition poses two unique challenges for deep UDA different from that in object classification. First, popular methods by the global alignment of source and target domain are no longer sufficient to acquire the discriminating power for classification in deep FR. Second, the face identities (classes) of source and target domain are non-overlapping, so that many skills developed in deep UDA which are used to further improve target performance based on sharing classes are inapplicable. In this sense, designing suitable adaptation method is the key to apply deep face recognition technique in ubiquitous scenes, but few research works has been done in this community.

In this paper, we propose a clustering-based domain adaptation (CDA) method for unconstrained face recognition. In order to address non-overlapping identities between domains, we introduce clustering algorithms into target domain to obtain pseudo-labels, by which the pre-learned model is adapted and the enhanced discriminative representations are learned. Specifically, CDA applies a simplified spectral clustering algorithm which requires neither overlapping classes nor the number of target classes. It generates pseudo-labels through a clustering graph where the nodes represent images and edges signify two images have larger cosine-similarity, and each connected component with at least three nodes in graph is saved as a cluster (identity). This scheme for domain adaptation is fundamentally different from the state-of-the-art methods which generate target pseudo-labels by maximum posterior probability of source classifier [24,31–33], these methods can not be utilized in FR due to non-overlapping classes of two domains.

To enhance the quality of clustering-based pseudo-labels, the proposed CDA method applies deep domain confusion network (DDC) [14] and deep adaptation networks (DAN) [15] to conduct global domain alignment before clustering, which optimize the learned representations by minimizing a measure of domain

discrepancy, i.e. maximum mean discrepancy (MMD). The hidden representations of images of different domain are embedded in a reproducing kernel Hilbert space, and the mean embeddings of distributions cross domains can be explicitly matched. Through utilizing MMD to optimize pre-learned model, DDC and DAN both alleviate the discrepancy between source and target face database and enhance model performance on target test data. Besides, with more transferable and generalized feature extracted from DDC and DAN, the calculated cosine-similarity of any two target images in our clustering algorithm is more accurate leading to higher quality of pseudo-labels. Comprehensive experiments are carried out in the GBU [11], IJB-A/B/C [12,34,35] and RFW [36] databases, significant performance gains are reached which indicates the competency of the proposed approach.

Our contributions can be summarized into three aspects.

- (1) We present a comprehensive study of scene adaptation in face recognition task, and empirically validate the necessity to perform deep domain adaptation. Even the deep models trained by large-scale training Web-collected data still fail to generalize well in many realistic scenes, such as those defined by Ugly data of GBU [11] and the low-quality data of IJB-A dataset [12]. This is caused by the mismatched distribution of training and testing data due to different illuminations, image quality, and shooting angles.
- (2) We propose a new clustering-based domain adaptation method to address a special domain adaptation task for face recognition where the training (source) and test (target) subjects are non-overlapping. CDA effectively learns the discriminative target feature by aligning the feature domain globally, and, at the meantime, distinguishing the target clusters locally. It first jointly applies DDC and DAN to reduce domain gap and learn domain-invariant representations, and thus provides more reliable underlying face representation for clustering. Then, a simplified spectral clustering method is proposed to generate pseudo-labels in the aligned feature space, and target discriminative representations are learned.
- (3) We perform extensive face recognition experiments by using the Web-collected dataset [10] as source domain, and GBU [11], IJB-A/B/C databases [12,34,35] as the target domains, and experimental results demonstrate the superiority of the proposed method. In particular, our method outperforms the state-of-the-art counterparts by a large margin on the

GBU dataset, although it is only based on the unsupervised adaptation from the target training data. Moreover, we also utilize our method to perform adaptation across races, and our CDA obtains promising performance on different races of RFW dataset [36].

The remainder of this paper is structured as follows. In the next section, we briefly review related work on deep FR and deep UDA. Then, we introduce the details of MMD and pseudo-labels in Section III. In Section IV, we introduce our clustering based domain adaptation algorithm in detail. Additionally, experimental results are shown and analyzed in Section V. Finally, we conclude and discuss future work.

2. Related work

2.1. Deep face recognition

In 2014, DeepFace [37] achieved the state-of-the-art accuracy on the famous LFW benchmark [8], approaching human performance on the unconstrained condition for the first time, by training a 9-layer model on 4 million facial images. Since then, research of FR focus has shifted to deep-learning-based approaches. More powerful loss functions are explored to learn deep discriminative features and are categorized into Euclidean distance based loss, angular/cosine margin based loss as well as softmax loss and its variations [38]. Euclidean distance based loss reduces intra-variance and enlarges inter-variance based on Euclidean distance. DeepID series [39–41] combined the face identification (softmax) and verification (contrastive loss) supervisory signals to learn a discriminative representation, and joint Bayesian (JB) was applied to obtain a robust embedding space. They trained 50 networks using a private dataset of 202,595 images and 10,117 subjects. FaceNet [42] used a triplet loss function aiming to separate the positive pair from the negative one by a distance margin and achieves good performance (99.63%) on LFW. VGG model [7] is a typical application based on VGGNet architectures [2]. It was trained on a large scale dataset of 2.6M images of 2622 subjects. Wen et al. [43] proposed a center loss to reduce the intra-class features variations. To separate samples more strictly and avoid misclassifying the difficult samples, angular/cosine margin based loss is proposed to make learned features potentially separable with a larger angular/cosine distance on a hypersphere manifold, such as Sphereface [44], L-softmax [45], Cosface [46], AMS [47] and Arcface [48]. In addition to Euclidean distance based loss and angular/cosine margin based loss, there are also many works taking effort to normalize feature or weight in softmax loss, e.g. L2-softmax [49] enforced all the features to have the same L2-norm, so that similar attention is given to good quality frontal faces and blurry faces with extreme pose; Ring loss [50] encouraged norm of samples being value R (a learned parameter) rather than explicit enforcing through a hard normalization operation.

Although these CNN based methods have achieved ultimate accuracy in LFW benchmark, they only focus on utilizing a massive amount of labeled facial images to train a CNN with strong generalization ability and testing on common benchmarks with same distribution. When there is domain shift and it is impossible to obtain labeled data in testing scenarios, the CNN pre-trained on the source data may not generalize well to target data.

2.2. Deep unsupervised domain adaptation

Mimicking the human vision system, domain adaptation is a particular case of transfer learning (TL) that utilizes labeled data in one or more relevant source domains to execute new tasks in a target domain [13]. Basically, the main challenge in domain

adaptation is the domain shift between the source domain and the target domain. To address this issue, in close-set DA where the images of the source and target domain are from the same set of categories, many UDA approaches are proposed and explore domain-invariant feature spaces by minimizing some measures of domain discrepancy such as statistic loss [14–18,51], adversarial loss [21–26]. MMD is a commonly-used statistic loss for UDA. The DDC proposed by Tzeng et al. [14] is optimized for classification loss in the source domain, while domain difference is minimized by one adaptation layer with the MMD metric. Long et al. [15] proposed DAN that matches the shift in marginal distributions across domains by adding multiple adaptation layers and exploring multiple kernels. Adversarial loss makes the distribution of both domains similar enough through domain classifier such that the network is fooled and can be directly used in the target domain. The domain-adversarial neural network (DANN) [22] integrated a gradient reversal layer (GRL) to train a feature extractor by maximizing the domain classifier loss and simultaneously minimizing the label predictor loss.

Besides, Zhang et al. [24,31–33,52] utilize the pseudo-labels to compensate the lack of categorical information and learn discriminative representations in the target domain. In [31], the idea of tri-training [53] was incorporated into domain adaptation. Two different networks assign pseudo-labels to unlabeled samples, another network is trained by these pseudo-labels to obtain target discriminative representations. Zhang et al. [24] iteratively selected pseudo-labeled target samples based on the classifier from the previous training epoch and re-trained the model by using the enlarged training set.

However, the assumption of close-set DA may not hold in real world application, and the source and target domain may not always share label space. Currently, open-set DA [54–58] is proposed to address this problem. In open-set DA, different domains only share partial classes and further contain their specific classes. Therefore, the key issue of open-set DA is to separate samples into shared and specific classes and align domains in shared label space. Cao et al. introduced a selective adversarial network (SAN) [54] to promote positive transfer by matching the data distributions in the shared label space via splitting the domain discriminator into many class-wise domain discriminators. Separate to Adapt (STA) [57] adopted a coarse-to-fine weighting mechanism to progressively separate the samples of unknown and known classes, and used instance-level weights to reject samples of unknown classes in adversarial domain adaptation. Zhang et al. [55] proposed a two domain classifier strategy to identify the importance score of source samples. Satio et al. [58] proposed a new adversarial learning method in which the feature generator can decrease or increase the probability for specific classes in order to align shared classes or reject specific classes. However, in face recognition, there is no shared class between source and target domain, which is a more complex and realistic setting compared to open-set DA. Domain shift in face recognition can not be addressed through simply aligning domains in shared label space.

2.3. Unsupervised domain adaptation for face recognition

In shallow face recognition, many UDA methods [29,30,59–61] were utilized to match the distributions of training and testing datasets. Yang et al. [59] developed a domain-shared group-sparse dictionary learning model to learn domain-shared representations with aligned joint distributions. Kan et al. [30] directly converted the source domain data to the target domain in the image space with the help of sparse reconstruction coefficients learnt in the common subspace. Zong et al. [62] learned a domain regenerator to regenerate the source and target samples by subspace learning and MMD, such that they can abide by the same or similar feature

distributions. Ni et al. [29] sampled several intermediate domains between the source and target domains, and represented each intermediate domain using a dictionary, then they applied invariant sparse codes across these domains to provide a shared feature representation which can be utilized for cross domain recognition. In deep learning era, deeper networks and larger unconstrained images are used to improve the performance of face recognition systems. However, deep FR is still affected by domain shift. Due to the unique challenges of deep FR, very few studies have focused on UDA for deep FR. Luo et al. [63] integrated the maximum mean discrepancies (MMD) estimator to CNN to decrease domain discrepancy. Sohn et al. [64] proposed an UDA method for video FR using large-scale unlabeled videos and labeled still images. They synthesized video frames from images by a set of transformations and utilized images, synthesized images, and unlabeled videos for domain adversarial training. A bi-shifting auto-encoder network (BAE) [65] is proposed to enforce the shifted source domain and target domain to share similar distribution, in which each sample of one domain can be sparsely reconstructed by several local neighbors from the other domain. Due to lack of labeled target data, these deep methods only align the feature domain globally, but ignoring the demand of discriminative ability on target domain. It is insufficient for deep FR, which is a fine-grained classification problem. We suggest that pseudo-labels are suitable to address this problem. However, pseudo-label based methods for object classification can not be used in FR because they all assume that there are shared classes between source and target domains and generate target pseudo-labels by maximum posterior probability of source classifier. In this paper, we propose a new clustering-based domain adaptation method to address this unique challenge.

3. Preliminary

In our case, we are given a set of labeled data from the source domain, and denote them as $\mathcal{D}_s = \{x_i^s, y_i^s\}_{i=1}^M$, where x_i^s is the i th source sample, y_i^s is its category label, and M is the number of source images. A set of unlabeled data from the target domain is given as well and is denoted as $\mathcal{D}_t = \{x_i^t\}_{i=1}^N$, where x_i^t is the i th target sample and N is the number of target images. The data distributions of two domains are different, $P(X_s, Y_s) \neq P(X_t, Y_t)$.

3.1. Maximum mean discrepancy

In the field of UDA, MMD [14,15] has been widely adopted as a standard distribution distance metric to measure the discrepancy between source and target domains. Given two distributions s and t , the MMD between them is defined as:

$$L_M(s, t) = \sup_{\|\phi\|_{\mathcal{H}} \leq 1} \|E_{x^s \sim s}[\phi(x^s)] - E_{x^t \sim t}[\phi(x^t)]\|_{\mathcal{H}}^2 \quad (1)$$

where E represents the expectation with regard to the distribution. ϕ represents the function that maps the original data to a reproducing kernel Hilbert space (RKHS). We have $MMD^2(s, t) = 0$ when s and t share the same distribution based on the statistic tests defined by MMD. The kernel functions which are associated with this mapping, $k(x^s, x^t) = \langle \phi(x^s), \phi(x^t) \rangle$, is defined as the convex combination of m PSD kernels k_u ,

$$\mathcal{K} = \left\{ k = \sum_{u=1}^m \beta_u k_u : \sum_{u=1}^m \beta_u = 1, \beta_u \geq 0, \forall u \right\} \quad (2)$$

where β_u is the coefficient of u th kernel and the commonly-used kernel is the Gaussian kernel $k_u(x^s, x^t) = e^{-\|x^s - x^t\|^2 / \gamma}$. Denote by $\mathcal{D}_s = \{x_i^s\}_{i=1}^M$ and $\mathcal{D}_t = \{x_i^t\}_{i=1}^N$ drawn from the distributions s and t ,

respectively, an empirical estimate of MMD is given as:

$$L_M(\mathcal{D}_s, \mathcal{D}_t) = \left\| \frac{1}{M} \sum_{i=1}^M \phi(x_i^s) - \frac{1}{N} \sum_{j=1}^N \phi(x_j^t) \right\|_{\mathcal{H}}^2 \quad (3)$$

The main idea of MMD-based network, i.e. DDC [14] and DAN [15], is to integrate MMD estimator to the CNN error so that the domain divergence is minimized. However, the formulation of MMD in Eq. (3) is computed in quadratic time complexity, it is prohibitively time-consuming for deep UDA. Gretton et al. [66] further suggested an unbiased approximation to MMD with linear complexity and it is suitable for gradient computation in a mini-batch manner:

$$L_M(\mathcal{D}_s, \mathcal{D}_t) = \frac{1}{M(M-1)} \sum_{i \neq j}^M k(x_i^s, x_j^s) + \frac{1}{N(N-1)} \sum_{i \neq j}^N k(x_i^t, x_j^t) - \frac{2}{MN} \sum_{i,j=1}^{M,N} k(x_i^s, x_j^t) \quad (4)$$

Through optimizing networks by MMD, the final classification decisions are made based on features that are invariant to the change of domains, i.e., have the same or very similar distributions in the source and the target domains, thus, the models trained on source data can generalize to target data.

3.2. Pseudo label

Pseudo-label is an alternative method for deep UDA in object classification assuming that source and target domain share the same classes [24,31–33,52]. CNN is trained supervised with source labeled data and is fine-tuned with target pseudo-labeled data that can be obtained by following steps. We denote $\{p_c(x_i^t)\}_{c=1}^{m_c}$ as the output from the Softmax layer of the source classifier in CNN, where each $p_c(x_i^t)$ is the probability that target sample x_i^t belongs to the c th classes, and m_c is the total number of classes. Then, the pseudo-label of x_i^t can be obtained by choosing the class with the maximum posterior probability:

$$\hat{y}_i^t = \arg \max_c p_c(x_i^t) \quad (5)$$

After that, the network is fine-tuned on pseudo-labeled target data with supervision of Softmax loss.

Furthermore, to suppress the negative influence of falsely-labeled samples, some studies are explored modified strategies which progressively select reliable pseudo-labels from the most confident predictions and re-train the model by using the enlarged training set. It can be formulated as follow:

$$\forall x_i^t \in \mathcal{D}_t, \omega_i = \begin{cases} 1, & \text{if } p_k(x_i^t) > \eta \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where $\mathcal{D}_t^{\omega_i=1}$ denotes the unlabeled target samples \mathcal{D}_t are partitioned into m_c classes. $\omega_i = 1$ indicates x_i^t to be selected in current training process; otherwise, x_i^t is not to be selected. η is a threshold which constrains the maximum posterior probability (confidence) of selected samples. η can be a constant, or a variable of the training step [32,52], or a variable of the classification accuracy of the current classifier measured by the labeled source data [24].

4. Clustering based domain adaptation

Due to the absence of labeled target samples, most deep DA methods for object classification, such as MMD, only align source and target domain globally. It is not effective enough and cannot ensure accuracy on the target domain in FR tasks where discriminative target representations are required. When lacking

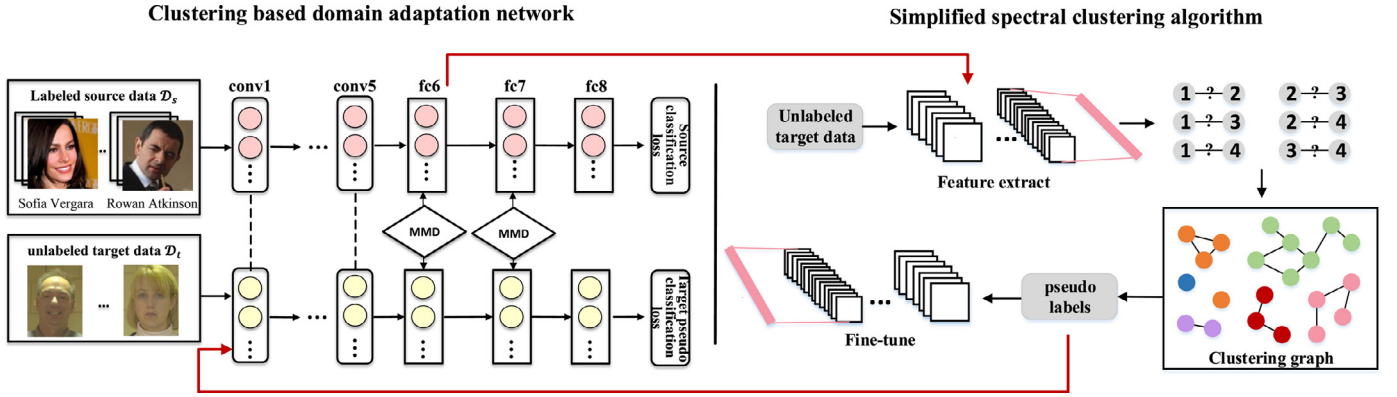


Fig. 2. The overall structure of the proposed method. **Left:** The clustering based domain adaptation network. Source classification loss supervises learning proceeds for source domain; MMD loss aims at minimizing the distribution discrepancy of two domains; target pseudo classification loss aims to learn discriminative target representations on pseudo-labels generated by clustering algorithms. Only using the first two losses to optimize networks is denoted as MMD-based networks. We first train a MMD-based network using labeled source data and unlabeled target data, then utilize target pseudo classification loss to further adapt target CNN after obtaining target pseudo-labels. **Right:** The simplified spectral clustering algorithm. With the target representations extracted by MMD-based network, a clustering graph is constructed where the nodes represent images and edges signify two images have larger cosine-similarity. Each connected component with at least three nodes is saved as a cluster (identity). Then, we can annotate the clustered nodes with pseudo labels and adapt the target CNN with them.

of target categorical information, we suggest that pseudo-labels [67] are suitable to address this problem, which encourages a low-density separation between classes in the target domain. However, adopting UDA in face recognition is a special domain adaptation task where the training (source) and test (target) subjects are non-overlapping, which means that traditional pseudo-labels based UDA methods relying on shared categories are inapplicable. To address this problem, we propose to introduce clustering algorithms into UDA. Many clustering algorithms are feasible for generating pseudo-labels in our clustering-based domain adaptation (CDA) network, and we design a simplified spectral clustering algorithm which is simple but effective for clustering faces in deep feature space. It clusters faces through connected subgraphs and can be adopted even if the number of target classes is large but unknown. The overall architecture of our method is depicted in Fig. 2.

4.1. Clustering algorithm

In this section, we formally introduce the detailed steps of simplified spectral clustering algorithm:

Compute similarity matrix. We feed unlabeled target data X_t into a deep model as input and extract deep features $\mathcal{F}(X_t)$. As we know, the clustering results depend not only on the choice of clustering algorithm, but also on the quality of the underlying face representation. Considering domain shift, the underlying target representation will not be perfect even using a strong source model. Therefore, the deep model here is pre-trained on source samples and further optimized by MMD to improve performance in target domain as much as possible. Then, with these deep representations, we construct a $N \times N$ similarity matrix, where N is the number of faces in target domain and entry at (i, j) , i.e. $s(i, j)$, is the cosine similarity between target representations $\mathcal{F}(x_i^t)$ and $\mathcal{F}(x_j^t)$.

Build clustering graphs. We consider two faces belonging to one identity if their cosine similarity is large. Thus, we can build a clustering graph $\mathcal{G}(n, e)$ according to similarity matrix, where the node n_i represents i -th target image and edge $e(n_i, n_j)$ signifies these two target images have larger cosine-similarity:

$$e(n_i, n_j) = \begin{cases} 1, & \text{if } s(i, j) > \alpha \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where α is the threshold for edges. Then, we simply save each connected component with at least p nodes as a cluster (identity) and the remaining images will be treated as scattered points. We

choose a minimum component size $p = 3$. Because the connected components with only one or two nodes may be the ones clustered incorrectly; even if this cluster is correct, low-shot class would deteriorate the long tail distribution of data. Furthermore, the threshold α is vital for clustering. If α is set to be lower, more faces of different identities will be clustered together which contains severe intra-class noise; otherwise, faces of one identity will split into more scattered points and be discarded, or they will split into smaller clusters leading to severe inter-class noise.

Pick up scattered points. Due to large variations, some images can not be clustered and be treated as scattered points. We pick up these scattered points by assuming that all samples of a given identity can be clustered around its corresponding prototype. The prototypes are computed by the average representation μ_k^t of all target samples in one cluster k obtained by connected component:

$$\mu_k^t = \frac{1}{|D_k^t|} \sum_{x_i^t \in D_k^t} \mathcal{F}(x_i^t) \quad (8)$$

where D_k^t is the set of all target images in k th cluster. Then, for each scattered point $x_{i(scatter)}^t$, we compute its cosine similarities with all prototypes, and add it to corresponding prototype with the largest cosine similarity. To obtain the samples with high confidence, we constrain that the similarity scores should above a certain threshold β :

$$\hat{y}_i^t = \begin{cases} \arg\max_k s_k, & \text{if } \max_k s_k > \beta \\ \infty, & \text{otherwise} \end{cases} \quad (9)$$

where, $s_k = \cos(\mathcal{F}(x_{i(scatter)}^t), \mu_k^t)$

So, we only cluster images with higher confidence to alleviate negative influence caused by falsely-labeled samples. Finally, we can annotate all clustered nodes with pseudo label \hat{y}_i^t , and adapt the network with supervision of Softmax loss.

4.2. Adaptation networks

We extend the VGGNet [2] and ResNet [4] architecture to our CDA network. As shown in Fig. 2, the architecture of our CDA consists of a source and target CNN, with shared weights. MMD estimators are adopted on higher layers of network which are called adaptation layers. We simply use a fork at the top of the network, after the adaptation layer. The inputs of source CNN are source labeled images while those of target CNN are target

unlabeled data. The goal of our approach is to minimize the following loss function:

$$L = L_S(X_s, y_s) + \lambda \sum_{l \in \mathcal{L}} L_M(D_s^l, D_t^l) + L_T(X_t, \hat{y}_t) \quad (10)$$

where the hyperparameter λ is a penalty parameter. D_s^l is the l th layer hidden representation for the source and target examples, and $L_M(D_s^l, D_t^l)$ (Eqn. (4)) is the MMD between the source and target evaluated on the l th layer representation. MMD loss makes the distributions of the source and target similar under the hidden representations. Selecting suitable adaptation layers can significantly enhance the transfer efficiency. According to the observation of [68], the transfer ability drops in higher layers with increasing domain discrepancy and transfer learning method would obtain better performance when transferring higher layers of the deep neural network. In CDA, we adopt multi-kernel MMD on the last two layers. $L_S(X_s, y_s)$ denotes source classification loss on the source data X_s and the ground truth labels y_s , which guarantees the performance of deep network. The third term, i.e. $L_T(X_t, \hat{y}_t)$, is our target pseudo classification loss on the target data X_t and the pseudo-labels \hat{y}_t , which learns more discriminative representations for target domain:

$$\begin{aligned} L_S(X_s, y_s) &= -\frac{1}{M} \sum_{i=1}^M \sum_{c=1}^{m_c} \mathbf{1}_{[c=y_i^s]} \log p_c(x_i^s) \\ L_T(X_t, \hat{y}_t) &= -\frac{1}{\hat{N}} \sum_{i=1}^{\hat{N}} \sum_{c=1}^{\hat{n}_c} \mathbf{1}_{[c=\hat{y}_i^t]} \log p_c(x_i^t) \end{aligned} \quad (11)$$

Here, we utilize Softmax loss (Arcface loss [48]) as our classification loss for source and target domain. In source classification loss $L_S(X_s, y_s)$, $\mathbf{1}_{[c=y_i^s]}$ is 1 when $c = y_i^s$, otherwise, it is 0; $p_c(x_i^s)$ is the probability that source sample x_i^s belongs to the c th classes, and m_c is the total number of source classes. The definition of target pseudo classification loss $L_T(X_t, \hat{y}_t)$ is similar to that of source classification loss where \hat{n}_c is the total number of target clusters and \hat{N} is the number of target samples clustered successfully.

4.3. Clustering based domain adaptation algorithm

The entire procedure of our method is depicted in Algorithm 1. In the first stage, the baseline model is trained with our source data, i.e. CASIA-Webface [10], so that we can use it as our source CNN. In the second stage, source classification loss and MMD loss are used to optimize MMD-based networks (i.e. DDC [14] and DAN [15]). MMD-based network is conducted by source CNN and weight-shared target CNN and is trained with the unlabeled target data and labeled source data so that the deep features are invariant to the change of domains, i.e., have the same or very similar distributions in the source and the target domains, and the performance of target domain is preliminarily improved. In the third stage, we extract deep features of target samples by MMD-based network, then adopt our clustering algorithms to generate pseudo-labels. Benefiting from better performance of MMD-based networks on target domain, the calculated cosine-similarity of any two target images in our clustering algorithm is more accurate leading to higher quality of pseudo-labels. In the forth stage, we adapt the target CNN on these pseudo-labeled target data with supervision of target pseudo classification loss. MMD-based networks address huge domain discrepancy to learn transferable representations for FR tasks and provide more reliable underlying face representation for clustering; while pseudo-labels encourage a low-density separation between target classes to learn more discriminative representations for FR tasks.

Algorithm 1 Clustering based domain adaptation algorithms.

Input:

Source domain labeled samples $\{x_i^s, y_i^s\}_{i=1}^M$, and target domain unlabeled samples $\{x_i^t\}_{i=1}^N$. Network learning rate μ , hyper parameter λ, α, β and p , network layer parameters Θ .

Output:

Network layer parameters Θ .

- 1: **Stage-1:** // Pre-train
 - 2: Train the baseline model on source labeled data;
 - 3: **Stage-2:** // MMD-adaptation
Adapt the network with MMD loss and source classification loss to learn domain-invariant representations and provide more reliable underlying face representation for clustering
 - 4: **Repeat:**
 - 5: $j = j + 1$
 - 6: Update the backpropagation error for x_i :

$$\frac{\partial L^j}{\partial x_i^{s(j)}} = \frac{\partial L_S^j}{\partial x_i^{s(j)}} + \lambda \frac{\partial L_M^j}{\partial x_i^{s(j)}}$$

$$\frac{\partial L^j}{\partial x_i^{t(j)}} = \lambda \frac{\partial L_M^j}{\partial x_i^{t(j)}}$$
 - 7: Update the network layer parameters Θ :

$$\Theta^{j+1} = \Theta^j - \mu^j \frac{\partial L^j}{\partial \Theta^j}$$

$$= \Theta^j - \mu^j \left(\sum_{i=1}^M \frac{\partial L^j}{\partial x_i^{s(j)}} \frac{\partial x_i^{s(j)}}{\partial \Theta^j} + \sum_{i=1}^N \frac{\partial L^j}{\partial x_i^{t(j)}} \frac{\partial x_i^{t(j)}}{\partial \Theta^j} \right)$$
 - 8: **Until convergence**
 - 9: **Stage-3:** // generate target pseudo labels by clustering algorithm
 - 10: Extract deep features of target unlabeled data and compute similarity matrix;
 - 11: Build clustering graphs according to Eqn. (7) and save each connected component with at least p nodes as a cluster;
 - 12: Add scattered points to corresponding clusters according to Eqn. (9);
 - 13: Annotate all clustered nodes with pseudo label \hat{y}_t^j .
 - 14: **Stage-4:** // Pseudo-adaptation
Adapt the network with target pseudo-labels using target pseudo classification loss to learn more discriminative target representations
 - 15: **Repeat:**
 - 16: $j = j + 1$
 - 17: Update the network layer parameters Θ :

$$\Theta^{j+1} = \Theta^j - \mu^j \frac{\partial L_T^j}{\partial \Theta^j} = \Theta^j - \mu^j \left(\sum_{i=1}^N \frac{\partial L_T^j}{\partial x_i^{t(j)}} \frac{\partial x_i^{t(j)}}{\partial \Theta^j} \right)$$
 - 18: **Until convergence**
-

5. Experiments

In this section, we evaluate our CDA method on five face recognition benchmarks, i.e. GBU [11], IJB-A/B/C [12,34,35] and RFW [36]. We will begin with introducing the detailed information and evaluation protocol of the datasets we utilized, followed by illustrating the training details of our experiments and presenting results and analyses.

5.1. Datasets and evaluation protocols

CASIA-WebFace: CASIA-WebFace dataset [10] is a large scale face dataset gathered from Internet. It contains 10,575 subjects and 494,414 images. The large scale of labeled facial data does great help to train CNNs. In our experiments, we adopt this dataset as the source domain data for training the classification network.

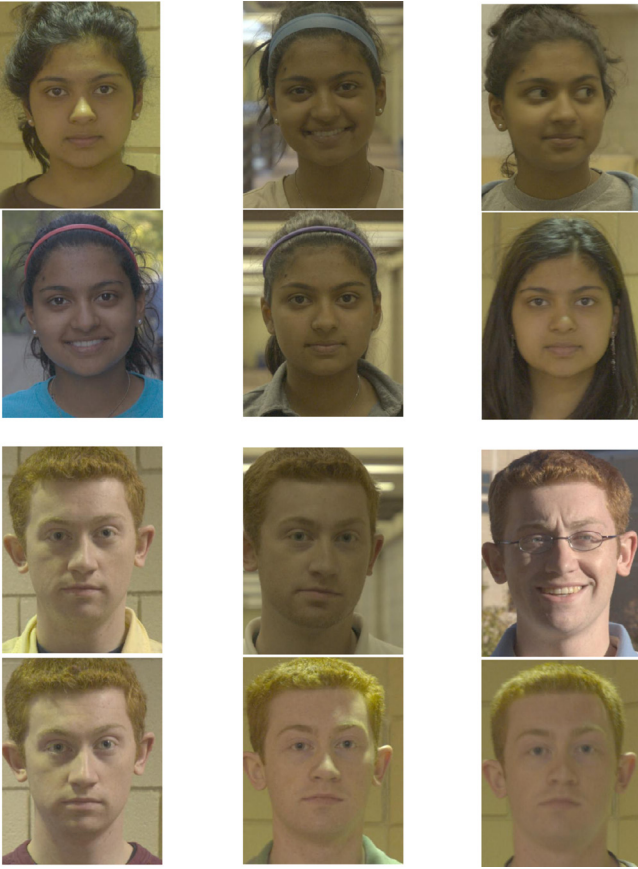


Fig. 3. Two example identities of the Good, Bad, and Ugly partition of GBU database. The top two rows show three pairs of images of the same person, sampled from the Good (left), Bad (middle), and Ugly (right) performance conditions. The second two rows show the same type of sample for a second person.

GBU: Its full name is *The Good, the Bad, and the Ugly Face Challenge* [11]. This dataset consists of three partitions, and different partitions contain pairs of images with different difficulty levels based on the performance of three top performers in the FRVT 2006. The Good partition consists of images which are easy to match; the Bad one contains pairs of average difficulty to recognize; the Ugly one contains pairs considered difficult. Fig. 3 shows three pairs of images of each person, sampled from the Good (left), Bad (middle), and Ugly (right) partition. This figure illustrates the variations in the appearance of a person across frontal images, e.g. different settings, expression and hairstyle. Each partition consists of a target set and a query set, and both them contain 1085 images of 437 distinct people. Following the evaluation protocol of [11], we use receiver operating characteristics (ROC) curve and the verification rate (VR) at a false positive rates (FAR) of 0.001 for each partition to compare the performances of different algorithms. In order to ensure that the subjects in target training set do not appear in target testing set, we utilize part of images from FRGC [69] (without label information) as the target training data, which consists of 19,270 still front faces.

IJB-A: IJB-A database [12] contains 5397 images and 2042 videos of 500 subjects, which are split into 20,412 frames, 11.4 images and 4.2 videos per subject. It is a joint face detection and FR dataset, in which both face detection and facial feature point detection are accomplished manually. The key characteristics of IJB-A are that it contains a mixture of images and videos in the wild and covers a full range of pose variations. IJB-A provides 10-split evaluations with two standard protocols, namely, face verification (1:1 comparison) and face identification (1:N search).

The performance of verification is reported using the true accept rates (TAR) vs. false positive rates (FAR) (i.e. ROC curve). The performance of identification is reported using the Rank-N (i.e. the cumulative match characteristic (CMC) curve) and the true positive identification rate (TPIR) vs. false positive identification rate (FPIR). There are ten random training (333 subjects) and testing (167 subjects) splits which occur at subject level, using all 500 IJB-A subjects. For each split, we adopt our CDA method by using its training data (without label information) as our target training data and using its testing data as our target testing data. The results are averaged over 10 testing splits.

IJB-B: The IJB-B dataset [34] is an extension of IJB-A [12], having 1845 subjects with 21.8K still images (including 11,754 face and 10,044 non-face) and 55K frames from 7,011 videos. The dataset is more challenging and diverse than IJB-A, with protocols designed to test detection, identification, verification and clustering of faces. Unlike the IJB-A dataset, it does not contain any training splits. We use images of IJB-A (without label information) as our target training data and use images of IJB-B as our target testing data.

IJB-C: The IJB-C dataset [35] is a further extension of IJB-B, having 3531 subjects with 31.3K still images and 117.5K frames from 11,779 videos. In total, there are 23,124 templates with 19,557 genuine matches and 15,639K impostor matches. Similar to IJB-B dataset, the protocols are designed to test detection, identification, verification and clustering of faces. The dataset also contains end-to-end protocols to evaluate the algorithms ability to perform end-to-end face recognition. We use images of IJB-A (without label information) as our target training data and use images of IJB-C as our target testing data.

RFW: Racial Faces in-the-Wild (RFW) dataset [36] is a testing database for studying racial bias in face recognition. Four testing subsets, namely Caucasian, Asian, Indian and African, are constructed, and each contains about 3000 individuals with 6000 image pairs for face verification. They can be used to fairly evaluate and compare the recognition ability of the algorithm on different races. We use RFW dataset to validate the effectiveness of our CDA method on transferring knowledge across races. In order to perform adaptation experiment, we utilize BUPT-Transferface dataset [36] to train our CDA model and test it on RFW. BUPT-Transferface dataset is a training dataset with four race subsets and is released with RFW. One training subset consists of about 500K labeled images of 10K Caucasians and three other subsets contain 50K unlabeled images of non-Caucasians, respectively. We use Caucasian as source domain and other races as target domains in our experiments.

5.2. Implementation details

For the baseline network, we employ the widely used VGGNet [2] and ResNet-34 [4] architecture. We finetune the VGG model [7] with the guidance of Softmax loss on the CAISA-Webface, and is called VGG(fin tune) model in our paper; while the ResNet-34 is trained with the guidance of Arcface loss [48] on the CAISA-Webface, and is called Arcface model in our paper.

For data processing of VGG, all the images of different datasets are aligned to the same reference point using three facial landmarks (left eye, right eye and center of mouth). The images are firstly resized to 250×250 and are then randomly cropped to 224×224 . We also augment the data by flipping it horizontally with 50% probability. And for data processing of ResNet, we use five facial landmarks for similarity transformation, then crop and resize the faces to 112×112 . Each pixel ($[0, 255]$) in RGB images is normalized by subtracting 127.5 and then being divided by 128.

For training CDA(vgg-soft) model, we select VGG model [7] which uses VGGNet [2] and is trained on VGGface dataset [7] and reports excellent results on LFW and YTF benchmarks.

However, we know nothing about the face aligned method in VGG model which may cause inconsistent alignment methods between training data and test data and thus results in a poor performance. To address this issue, We use the fine-tuning architecture similar to [14,15] where CASIA-WebFace dataset [10] is utilized as source data to fine-tune the VGG model. The CASIA-WebFace dataset and other target datasets share the uniform alignment methods as we mentioned before. The based learning rate is fixed at 10^{-4} . As the last classifier is trained from scratch, we set its learning rate to be 10 times that of the lower layers. The batch size is set to 32 and the network is trained for 2×10^4 iterations.

After fine-tuning the VGG model with our source data, we utilize the unlabeled target data and labeled source data to adapt the baseline network by MMD. Our network architecture is comprised of two basic CNNs which are identical in structure and shared by parameters. One is for classification on source data and the other is for representation learning on target data. We use Softmax loss as source classification loss and fix the learning rate of all layers to 10^{-4} . The hyper-parameter λ in Eq. 11 is fixed at 0.5. The kernel in

MMD is Gaussian kernel $k(x^s, x^t) = e^{-\|x^s - x^t\|^2 / \gamma}$ where γ donates the bandwidth. In our experiments, DAN(vgg-soft) [15] applies multi-kernel MMD on both *fc6* and *fc7* layer. Five Gaussian kernels are utilized by setting bandwidth to $\gamma_m \cdot (1, 2^1, 2^2, 2^3, 2^4)$ where γ_m is set to the median pairwise distances [70] on training data. DDC(vgg-soft) [14] adopts single-kernel MMD on *fc7* layer, and it only utilizes one Gaussian kernel in which bandwidth is set to γ_m . To evaluate the effectiveness of multi-layer and multi-kernel adaptation more comprehensively, we further make several variants of MMD-based network, namely single-kernel MMD on both *fc6* and *fc7* layer and multi-kernel MMD on *fc7* layer. We denote them as $DDC_{ml}(\text{vgg-soft})$ and $DDC_{mk}(\text{vgg-soft})$, respectively.

For our clustering methods, the hyper-parameter p is set to be 3. We set the parameter α and β in Eq. 7 and Eq. 9 as 0.675 and 0.8 in CASIA \rightarrow GBU task, and set them as 0.65 and 0.8 in CASIA \rightarrow IJB-A/IJB-B/IJB-C task. After obtaining the pseudo-labels, we further fine-tune the target network with them. We use Softmax loss as target pseudo classification loss. The learning rate is started from $1e-4$ and decreased twice with a factor of 10 when errors plateau. The network is trained for 2×10^4 iterations. We set the batch size, momentum, and weight decay as 64, 0.9 and $5e-4$, respectively.

For training CDA(res-arc) model, we first train a Arcface model with the guidance of Arcface loss [48] on the CAISA-Webface. We set the batch size, momentum, and weight decay as 200, 0.9 and $5e-4$, respectively. The learning rate is started from 0.1 and decreased twice with a factor of 10 when errors plateau. After that, we utilize the unlabeled target data and labeled source data to adapt Arcface model by MMD. We use Arcface loss as source classification loss and fix the learning rate of all layers to $1e-3$. The hyper-parameter λ in Eq. (11) is fixed at 5. DAN(res-arc) [15] applies multi-kernel MMD on last two fully-connected layers. For our clustering methods, we set the parameter α and β in Eqs. (7) and (9) as 0.8 and 0.85 in CASIA \rightarrow GBU task, and set them as 0.7 and 0.85 in CASIA \rightarrow IJB-A/IJB-B/IJB-C task. After obtaining the pseudo-labels, we further fine-tune the target network with them. We use Softmax loss as target pseudo classification loss. The learning rate is $1e-3$. We set the batch size, momentum, and weight decay as 200, 0.9 and $5e-4$, respectively. Other experimental settings are similar to CDA(vgg-soft).

5.3. Experiment results

CASIA \rightarrow GBU. In the experiment of GBU dataset [11], we report the verification rate at a FAR of 0.001 and ROC curve for three partitions, i.e. the Good, the Bad and the Ugly. Fusion method in [6] denotes the FRVT 2006 fusion algorithm and the result VGG

Table 1

VR at FAR of 0.001 for GBU partitions [11].

Method	Ugly	Bad	Good
LRPCA-face [11]	7.00%	24.00%	64.0%
Fusion [6]	15.00%	80.00%	98.00%
VGG [6]	26.00%	52.00%	85.00%
Arcface ^a [48]	75.00%	90.32%	96.21%
VGG(finetime) ^b	48.80%	73.55%	95.57%
DDC(vgg-soft) ^c [14]	60.90%	86.68%	98.24%
DDC _{ml} (vgg-soft) ^c	63.42%	87.08%	98.54%
DDC _{mk} (vgg-soft) ^c	68.42%	87.68%	98.67%
DAN(vgg-soft) ^c [15]	69.42%	88.87%	98.93%
CDA(vgg-soft) (ours)	73.58%	92.93%	99.18%
CDA(res-arc) (ours)	83.96%	94.84%	97.81%

^a Arcface is one of our baseline networks. It uses ResNet-34 architecture and is trained with the guidance of Arcface loss [48] on the CAISA-Webface.

^b VGG(finetime) is one of our baseline networks. It finetunes the VGG model [7] supervised with Softmax on CASIA-WebFace dataset.

^c DDC, DDC_{ml}, DDC_{mk} and DAN represent the variants of MMD-based network.

was reported in [6] by utilizing the VGG model [7]. The LRPCA-face model is a baseline algorithm in GBU dataset [11] which is a refined implementation of the standard PCA-based FR algorithm. The VGG(finetime) represents one of our baseline networks which finetunes the VGG model with CASIA-WebFace dataset [10]; and Arcface is the other baseline network which uses ResNet-34 architecture and is trained with the guidance of Arcface loss [48] on the CAISA-Webface. The exact results are shown in Table 1 and Fig. 4.

From the results, we can see several important observations. (1) For Ugly partition, all the models give the accuracies of less than 84% and specially an extremely low total accuracy of 15% with Fusion model, showing face verification on Ugly partition is a very challenging task despite of its frontal faces. Significantly, the performance of deep models is unsatisfactory as well and VGG only achieves 26% on Ugly partition, which illustrates the limitation of existing deep models trained with Web-collected dataset and the necessity of adopting UDA in FR tasks. (2) Compared with the results of VGG reported in [6], our baseline model fine-tuned VGG with CASIA-WebFace [10] obtains much better performance, which improves the accuracy to 48.80%, 73.55%, 95.57% on Ugly, Bad and Good partition. The results suggest that the uniform face aligned algorithm of training and testing data is the key to ensure performance in the FR problem. (3) MMD-based networks, i.e. DDC(vgg-soft)[14], DDC_{ml}(vgg-soft), DDC_{mk}(vgg-soft) and DAN(vgg-soft) [15], substantially outperform VGG(finetime) model on target dataset. This confirms that incorporating MMD to deep networks and minimizing the domain discrepancy are really helpful. (4) Single-kernel MMD models (DDC(vgg-soft) and DDC_{ml}(vgg-soft)) obtain a little bit worse results compared with multi-kernel MMD (DDC_{mk}(vgg-soft) and DAN(vgg-soft)). It is because multiple kernels with different bandwidths can match both the low-order moments and high-order moments resulting in a better alignment of distribution of source and target domain. (5) The DAN(vgg-soft) obtains the best performances compared with other MMD-based networks, which superior to our VGG(finetime) by about 20.62% on the Ugly, 14.13% on the Bad and 3.1% on the Good. In addition to multi-kernel adaptation, DAN(vgg-soft) is also benefited from multi-layer adaptation. In deep networks, representations of different layers correspond to different levels of abstraction, changing from low-level primary elements to multifarious facial attributes. Hence the hidden representations of all the task-specific layers need to be matched to consolidate the adaptation quality at all levels. (6) When introducing clustering algorithms and pseudo-labels into DAN(vgg-soft) models, the performances of our CDA(vgg-soft) method further improve and obtain the best performances with 73.58%, 92.93% and 99.18% for Ugly, Bad and Good

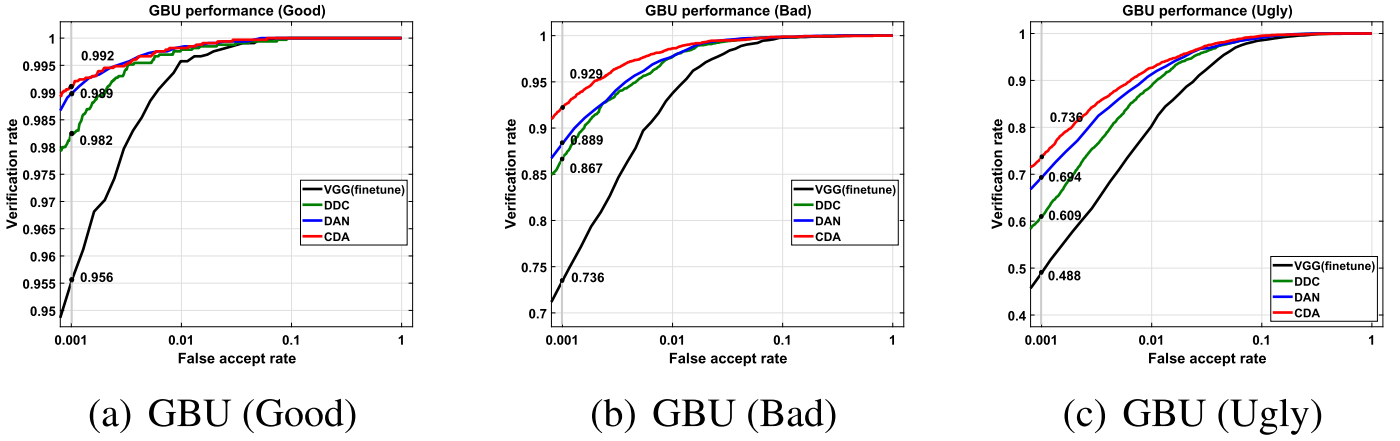


Fig. 4. The ROC curves on (a) Good, (b) Bad and (c) Ugly partition of GBU database. Black lines are ROC curves of *VGG(finetune)* model; Green lines are those of *DDC(vgg-soft)* model; Blue lines are those of *DAN(vgg-soft)* model; Red lines are those of our *CDA(vgg-soft)* model. The verification rate for each partition at a FAR of 0.001 is highlighted by the vertical lines at FAR=0.001.

Table 2

Performance evaluation on the IJB-A dataset [12]. The results are averaged over 10 testing splits.

Method	IJB-A Verification TAR			IJB-A Identification TPIR			
	FAR = 0.001	FAR = 0.01	FAR = 0.1	FPIR = 0.01	FPIR = 0.1	Rank-1	Rank-10
Bilinear-CNN [71]	–	–	–	14.20%	34.10%	58.80%	–
Face-Search [72]	–	73.30%	–	38.30%	61.30%	82.00%	–
Deep-Multipose [73]	–	78.70%	–	52.00%	75.00%	84.60%	94.70%
Triplet-Similarity [74]	–	79.00%	–	55.60%	75.41%	88.01%	97.38%
Joint Bayesian [75]	–	83.80%	–	57.68%	78.97%	90.30%	97.70%
VGG [7]	64.19%	84.02%	96.09%	47.37%	74.30%	91.11%	98.25%
Arcface ^a [48]	74.19%	87.11%	94.87%	65.36%	80.71%	90.68%	96.07%
VGG(finetune) ^b	67.96%	84.78%	95.80%	56.36%	76.05%	92.61%	98.54%
DDC(vgg-soft) [14]	72.78%	86.80%	96.34%	61.71%	80.02%	92.93%	98.81%
DDC _{ml} (vgg-soft) ^c	72.97%	87.74%	96.70%	62.82%	81.30%	92.91%	98.62%
DDC _{mk} (vgg-soft) ^c	72.53%	87.13%	96.54%	61.58%	82.33%	92.54%	98.52%
DAN(vgg-soft) [15]	72.88%	87.20%	96.34%	62.81%	81.54%	92.47%	98.33%
CDA(vgg-soft)(ours)	74.76%	89.76%	98.19%	66.85%	85.32%	94.89%	99.23%
CDA(res-arc)(ours)	82.45%	91.11%	96.96%	75.49%	87.76%	93.61%	97.62%

^a Arcface is one of our baseline networks. It uses ResNet-34 architecture and is trained with the guidance of Arcface loss [48] on the CAISA-Webface.

^b VGG(finetune) is one of our baseline networks. It finetunes the VGG model [7] supervised with Softmax on CASIA-WebFace dataset.

^c DDC_{ml} adopts single-kernel MMD on both *fc6* and *fc7* layer and DDC_{mk} adopts multi-kernel MMD on *fc7* layer.

set. We can draw conclusions that only aligning the feature space through MMD is not enough for FR and that further learning target discriminative representations using pseudo-labels is an effective way to boost the performance. Moreover, the results quantitatively prove the good quality of pseudo-labels generated by our clustering method. (7) Without adaptation, Arcface [48], which published in CVPR'19 and reported SOTA performance on the LFW and MegaFace challenges, can not obtain perfect performance on GBU due to domain gap. Our CDA(res-arc) can outperform Arcface method and even achieve about 3% gains on Ugly partition.

CASIA → IJB-A. We perform experiments in two settings on the IJB-A benchmark dataset [12]: the TAR at different FAR of 0.1, 0.01, and 0.001 for verification; the TPIR at different FPIR of 0.1, 0.01 and the rank-1, rank-10 accuracy for identification. Table 2 and Fig. 5 report the results of face verification and identification. We can observe that the VGG model does not perform well on IJB-A benchmarks. Benefiting from the same aligned method of training and testing data, VGG(finetune) model obtains a little promotion compared to VGG model, but its performance is still imperfect. The images and video frames in IJB-A dataset [12] contains full pose variation and a wide variation in imaging conditions and geographic origin. It is challenging for models trained with VG-Gface database [7] or CASIA-Webface databases [10] due to large domain gap. For example, video frames in IJB-A database are

likely to be degraded for motion or out-of-focus blur, compression noise or scale variations. When we reduce their domain gap using MMD-based networks, the improvement becomes more significant. Especially, DAN(vgg-soft) boosts around 9% TAR@FAR=0.001 for verification, and around 15% FNIR@FPIR=0.01 for identification compared with VGG model. It proves that the source networks trained with frontal and high-definition faces can adapt to recognize the blur images of large pose variations to a certain extent through domain adaptation. Similar to the experiments on GBU, multi-layer MMD also attains higher accuracy than single-layer MMD in most cases, which confirms the capability of multi-layers for distribution adaptation. After introducing clustering algorithms and pseudo-labels into DAN(vgg-soft), the CDA(vgg-soft) model surpasses other methods and outperforms DAN(vgg-soft) by about 2–4% on all metrics, which further demonstrates the advantage of our clustering algorithms. Further, when compared with the SOTA methods, i.e. Arcface, our CDA(res-arc) can still obtain better performance.

CASIA → IJB-B/C. We perform experiments in two settings on the IJB-B and IJB-C benchmark dataset [34,35]: the TAR at different FAR of 0.1, 0.01, and 0.001 for verification; the rank-1 and rank-10 accuracy for identification. Table 3 reports the results of face verification and identification. We compare our proposed method with Government-off-the-shelf (GOTS-1 [34]), Bodla et al. [76], VGG [7] and Arcface [48] on IJB-B dataset; and compare our

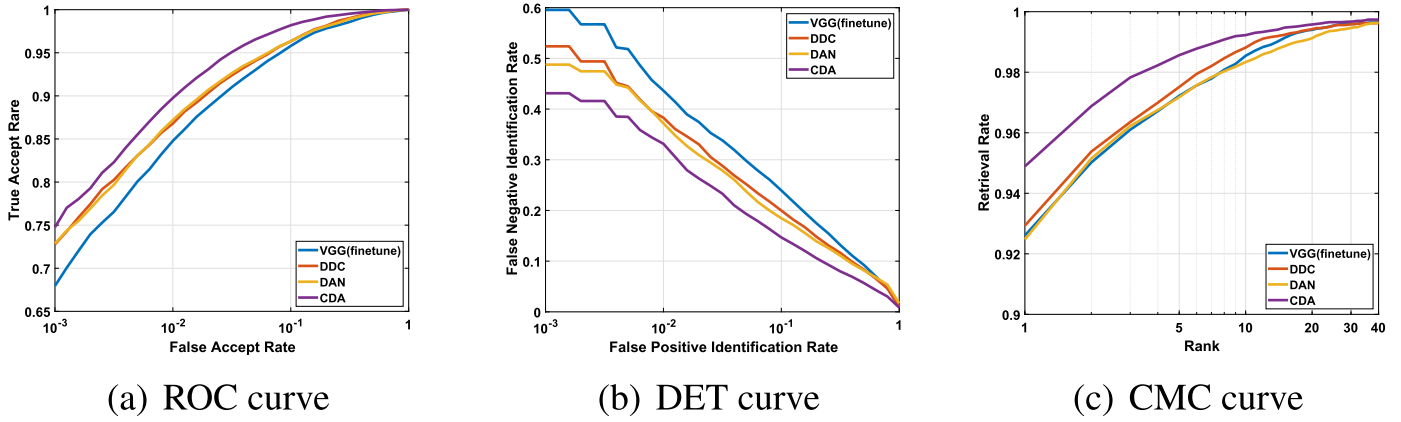


Fig. 5. Results on the IJB-A dataset (average over 10 splits). (a) ROC curve for the compare protocol (higher is better). (b) DET curve for the search protocol (lower is better). (c) CMC curve for the search protocol (higher is better).

Table 3

Performance evaluation on the IJB-B [34] and IJB-C [35] dataset.

Method	IJB-B					IJB-C				
	Verification TAR@FAR			Identification		Verification TAR@FAR			Identification	
	0.001	0.01	0.1	Rank-1	Rank-10	0.001	0.01	0.1	Rank-1	Rank-10
GOTS-1 [34]	33.00%	60.00%	78.00%	42.00%	62.00%	–	–	–	–	–
GOTS-2 [35]	–	–	–	–	–	32.00%	62.00%	80.00%	–	–
FaceNet [42]	–	–	–	–	–	66.00%	82.00%	92.00%	–	–
DR-GAN [77]	–	–	–	–	–	66.10%	82.40%	–	70.80%	82.80%
VGG [7]	72.00%	86.00%	–	78.00%	89.00%	75.00%	86.00%	95.00%	–	–
Bodla et al. [76]	83.00%	92.50%	–	–	–	–	–	–	–	–
Yin et al. [78]	–	–	–	–	–	75.60%	89.20%	–	77.60%	86.10%
Arcface ^a [48]	86.11%	93.40%	97.66%	86.43%	93.33%	88.88%	94.76%	98.10%	88.05%	93.56%
CDA(res-arc) (ours)	87.35%	94.55%	98.08%	86.22%	93.33%	88.06%	94.85%	98.33%	88.19%	93.70%

^a Arcface here is our baseline network which uses ResNet-34 architecture and is trained with the guidance of Arcface loss [48] on the CAISA-Webface.

method with GOTS-2 [35], FaceNet [42], DR-GAN [77], Yin et al. [78], VGG [7] and Arcface [48] on IJB-C dataset. From the results, we can see that our *CDA(res-arc)* achieves improvement over the previous SOTA methods, i.e. Arcface, with TAR of 87.35% at FAR = 10e-3 on IJB-B; while on IJB-C, it achieves a Rank1 accuracy of 88.19% in face identification. In our *CDA*, MMD-based networks address huge domain discrepancy to learn transferable representations and provide more reliable underlying face representation for clustering; while pseudo-labels further learn more discriminative representations for FR tasks. Actually, in our experiments, we just utilized limited number of images in IJB-A as target training data to achieve such improvement on these two challenging benchmarks. If more target training data are used to adapt source model, more significant improvement can be obtained.

Caucasian → Non-Caucasian. Some papers [36,79] have proved that existing face recognition algorithms indeed suffer from racial bias. Due to the domain gap among different races, training and testing on different races results in severe performance drop. To validate the effectiveness of our domain adaptation method, we adopt *CDA* to transfer knowledge among different races. We use BUPT-Transferface as training data, and use RFW [36] as testing data. Labeled Caucasians are utilized as source domain and unlabeled Indians/Asians/Africans are utilized as target domains in our experiments. The results are given in Table 4 and we have the following observations. (1) The Softmax and Arcface model which are trained on Caucasians perform well on Caucasian testing subset, but the accuracy drops on Asian and African because of domain gap. For example, the accuracy of the ArcFace model on caucasian testing subset reaches 94.78%, but its accuracy dramatically decreases to less than 85.13% on Asian subset. (2) *DDC(res-soft)*

Table 4

Verification accuracy (%) on 6000 pairs of RFW dataset [36]. “(res-soft)” represents the ResNet-34 methods using Softmax as source classification loss; while “(res-arc)” represents the ones using Arcface.

Methods	Caucasian	Indian	Asian	African
Softmax ^a	94.12%	88.33%	84.60%	83.47%
<i>DDC(res-soft)</i> [14]	–	90.53%	86.32%	84.95%
<i>DAN(res-soft)</i> [15]	–	89.98%	85.53%	84.10%
<i>CDA(res-soft)</i> (ours)	–	90.73%	88.88%	87.42%
Arcface ^a [48]	94.78%	90.48%	86.27%	85.13%
<i>DDC(res-arc)</i> [14]	–	91.63%	87.55%	86.28%
<i>DAN(res-arc)</i> [15]	–	91.78%	87.78%	86.30%
<i>CDA(res-arc)</i> (ours)	–	92.08%	88.80%	88.12%

^a Softmax and Arcface here are our baseline networks which use ResNet-34 architecture trained on the CAISA-Webface.

[14] and *DAN(res-soft)* [15] align Caucasian domain and other race domain with help of MMD. But they are only superior to baseline by about 1–2% which confirms our thought that only aligning domains globally is not enough for face recognition. (3) When adopting clustering algorithms and pseudo-labels, our *CDA(res-soft)* and *CDA(res-arc)* model outperform the baseline models, especially *CDA(res-arc)* obtains the best performances with 92.08%, 88.80% and 88.12% on Indian, Asian and African set.

5.4. Empirical analysis

Feature visualization. To demonstrate the transferability of the MMD learned features, the visualization comparisons are conducted at feature level. First, we randomly extract the deep

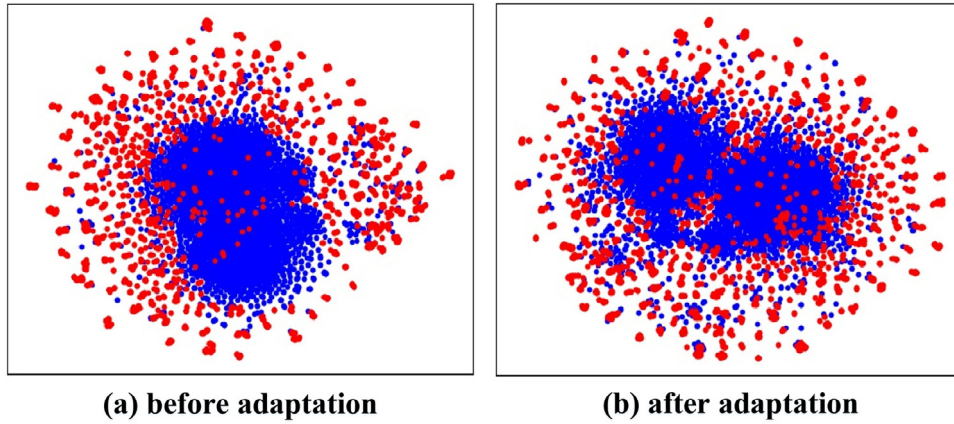


Fig. 6. Feature visualization. We confirm the effects of MMD through a visualization of the learned representations using t-distributed stochastic neighbor embedding (t-SNE) [80]. Blue points are source samples and red are target samples. (a) are trained without any adaptation, (b) are trained with MMD method. As we can see, compared to non-adapted method, MMD method can help our CDA to align the source features and target features to a certain extent and improve the performance of target domain.

features of 5000 source and 5000 target images in task CASIA \rightarrow GBU (Ugly) with VGG(*finetune*) model and DAN(vgg-soft) model, respectively. The features are visualized using t-distributed stochastic neighbor embedding (t-SNE) [80], as shown in Fig. 6. Fig. 6(a) shows the representations without any adapt. As we can see, the distributions are separated between domains, which visually proves that there is domain gap between images of CASIA-Webface [10] and GBU database [11]. Fig. 6(b) shows the result for DAN(vgg-soft) method where features are aligned to some extent. More source and target data begin to mix in feature space so that there is not a clear boundary between them. Therefore, we conclude that the MMD does help our CDA(vgg-soft) to minimize domain discrepancy and align feature space between source and target domain so that the performance of target domain improves. However, due to the particularity of face data, e.g. a larger number of identities as well as non-overlapping identities of source and target domain, misalignment still exists even after adaptation. It also verifies that MMD-adaptation is not enough for face recognition.

Parameter sensitivity. Besides the MMD penalty parameter λ , our clustering method involves another vital parameter α in Eq. (7) which controls the connection of edges in graph. Two target nodes will be connected to each other in our graph only if their cosine-similarity is larger than α . To have a closer look at this parameter, we perform sensitivity analysis for it in transfer tasks CASIA \rightarrow GBU (Ugly) by varying the parameter of interest in {0.6, 0.625, 0.65, 0.675, 0.7}. We generate different target pseudo-labels according to different parameter α , then fine-tune the target CNN with them respectively. The fine-tuning results are shown in

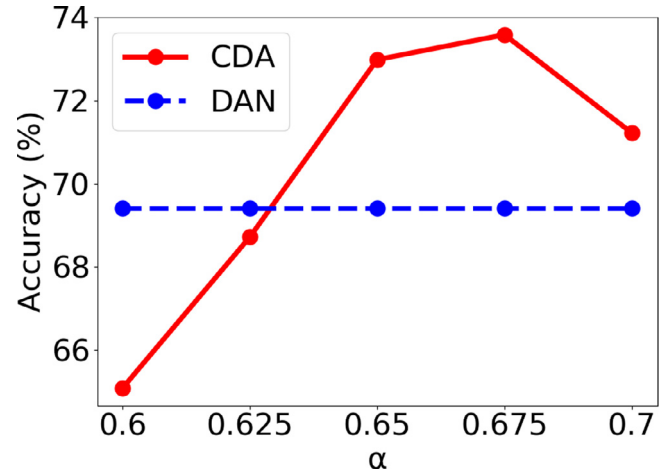


Fig. 7. Parameter sensitivity of α (dashed lines show best DAN(vgg-soft) results).

Fig. 7, with the results of DAN(vgg-soft) shown as dashed lines. We observe that the accuracy first increases and then decreases as α varies and demonstrates a desirable bell-shaped curve. This justifies our assumption that the parameter α in Eq. (7) makes a tradeoff between intra-noise and inter-noise of generated pseudo-labels. If α is set to be lower, more faces of different identities will be clustered together which contains severe intra-class noise; otherwise, faces of one identity will split into more scattered

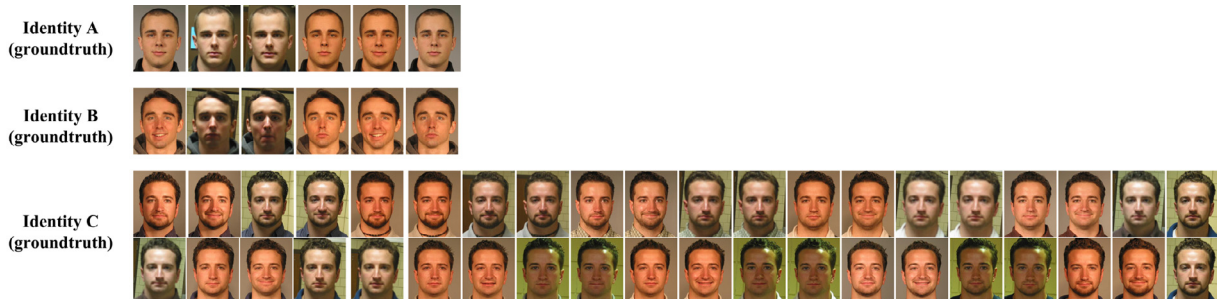


Fig. 8. Three examples of “pure” clusters generated by clustering method on our target training set of GBU. In top two rows, each row shows the images of one identity; the bottom two rows are images belong to the third identity. For each identity, all images in training set are grouped into one cluster together perfectly.

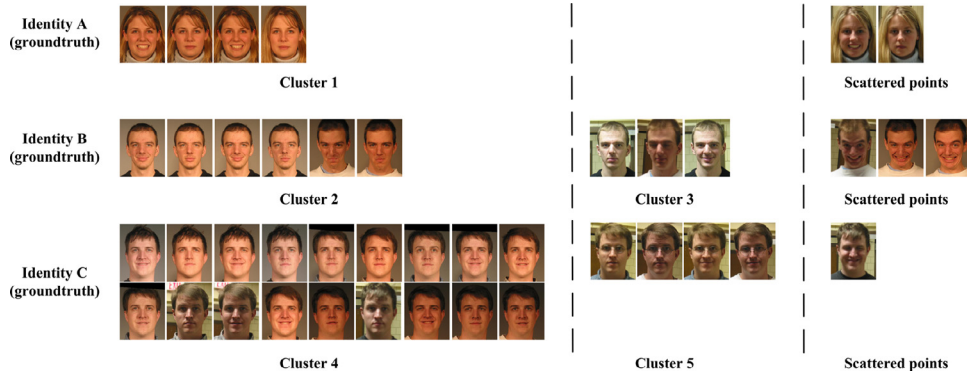


Fig. 9. Three examples of “split” clusters generated by clustering method on our target training set of GBU. In top two rows, each row shows the images of one identity; the bottom two rows are images belong to the third identity. For the first identity, partial images are clustered together, i.e. *cluster1*, but remaining images are treated as scattered points and are discarded. For the second and third identity, the images are split into some scattered points and two clusters, which leads to inter-noise.

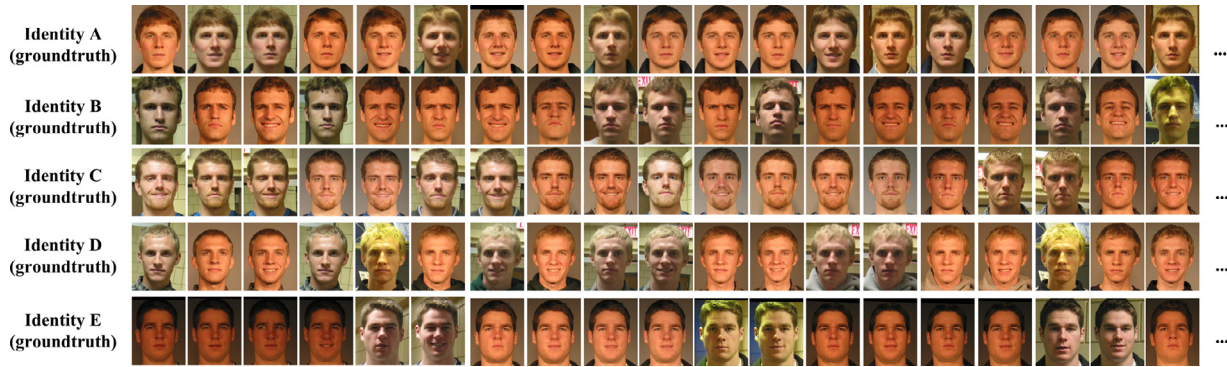


Fig. 10. One example of “impure” cluster generated by clustering method on our target training set of GBU. Each row shows the images of one identity but all images of these five identities are clustered together incorrectly.

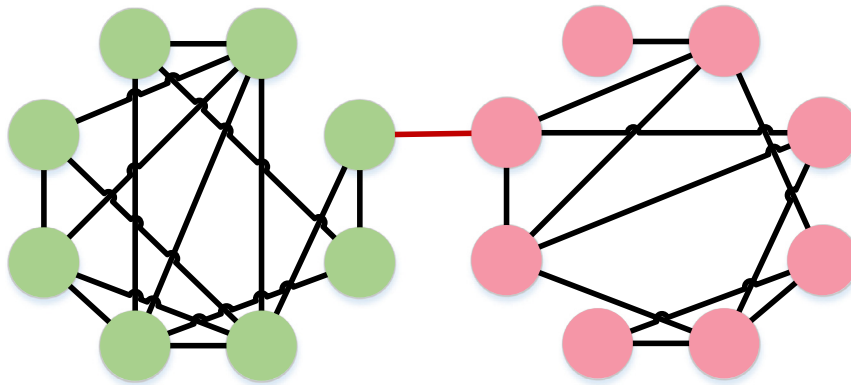


Fig. 11. The explanation of the existence of “impure” cluster. The green and pink points denote the images of two different identities in one cluster. When there happen to be two or more images of different identities looked like each other, they will be connected in our clustering graph, i.e. the red line, so the images of these people are grouped into one cluster when we generate pseudo-labels through connected component.

points and be discarded, or they will split into smaller clusters leading to severe inter-class noise.

Examples of clustering. As we know, the results of adaptation depend on the quality of pseudo-labels generated by our clustering algorithms. To visually evaluate our clustering method, we show some example clusters on our target training set of GBU in Figs. 8–10. Fig. 8 shows “pure” clusters which contain neither intra-noise nor inter-noise, that is to say, all images of one identity are grouped into one cluster together perfectly even if there are variations in expression, lighting, hairstyle, etc. In Fig. 9, examples of “split” clusters are presented. Although reliable cluster, e.g. *cluster2*, is formed with partial images of one identity, remaining

images are treated as scattered points or are split into another different clusters, e.g. *cluster3*, which results in inter-noise. This phenomenon usually occurs due to large variations. Fig. 10 shows example of “impure” cluster in terms of subject identity. Five different individuals are grouped into one cluster leading to serious intra-noise. When going deep into this type of clusters, we find that it usually happens to the identities whose images’ number is quite large. We give the explanation of this phenomenon in Fig. 11. A larger number of images per identity increase the probability of connectivity of different people in our clustering algorithms. Among massive images of two people, there happen to be two or more images of different identities looked like each other and

their cosine-similarities are larger than the parameter α . Even if two similar images, they will be connected in our clustering graph so the images of these people are grouped into one cluster when pseudo-labels are generated through connected component.

6. Conclusion

In this paper, we focus on the issue of domain discrepancy between source training data and target testing data in face recognition scenario. We address it in the viewpoint of unsupervised domain adaptation. First, considering the special problems of non-overlapping classes between two domains in FR, we further propose to introduce clustering algorithms into UDA to obtain pseudo-labels in the deep feature space, and design a simplified spectral clustering algorithm which requires neither overlapping classes between two domains nor the number of target classes. Second, to minimize domain discrepancy and enhance the quality of clustering-based pseudo-labels, we introduce deep UDA methods, namely DDC and DAN. Our CDA method effectively learns the discriminative target feature by aligning the feature domain globally, and, at the meantime, distinguishing the target clusters locally. Comprehensive experiments are carried out in the GBU and IJB-A/B/C databases, significant performance gains are reached which indicates the competency of the proposed approach.

In terms of future work, (1) while the underlying face representation we employ in clustering method works reasonably well for unconstrained face images, it could still be improved in a number of ways (e.g., selecting more reliable source training sets, or improving the transferability of deep model). (2) While we were able to boost the performance of target testing data, the quality of pseudo-labels still needs to be improved. So designing a better clustering method for UDA is a vital problem to be done in FR task. (3) We consider to use the “easy-to-hard” scheme which progressively selects reliable pseudo-labeled target samples from the most confident predictions or utilize the training skills of noisy data to alleviate the negative influence of falsely-labeled samples.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRedit authorship contribution statement

Mei Wang: Conceptualization, Methodology, Software, Data curation, Writing - original draft, Visualization, Investigation. **Weihong Deng:** Supervision, Software, Validation, Writing - review & editing, Funding acquisition.

Acknowledgments

This work was partially supported by the [National Natural Science Foundation of China](#) under Grant Nos. 61871052 and 61573068.

References

- [1] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: *Proceedings of the NIPS*, 2012, pp. 1097–1105.
- [2] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv:1409.1556 (2014).
- [3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, et al., Going deeper with convolutions, in: *Proceedings of the CVPR*, 2015.
- [4] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the CVPR*, 2016, pp. 770–778.
- [5] J. Hu, L. Shen, G. Sun, Squeeze-and-Excitation Networks, arXiv:1709.01507 (2017).
- [6] P.J. Phillips, A cross benchmark assessment of a deep convolutional neural network for face recognition, in: *Proceedings of the Twelfth IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, IEEE, 2017, pp. 705–710.
- [7] O.M. Parkhi, A. Vedaldi, A. Zisserman, et al., Deep face recognition, in: *Proceedings of the BMVC*, 1, 2015, p. 6.
- [8] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: a database for studying face recognition in unconstrained environments, Technical Report, 07–49, University of Massachusetts, Amherst, 2007.
- [9] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, in: *Proceedings of the CVPR*, IEEE, 2011, pp. 529–534.
- [10] D. Yi, Z. Lei, S. Liao, S.Z. Li, 2014. arXiv: 1411.7923
- [11] P.J. Phillips, J.R. Beveridge, B.A. Draper, G. Givens, A.J. O’Toole, D. Bolme, J. Dunlop, Y.M. Lui, H. Sahibzada, S. Weimer, The good, the bad, and the ugly face challenge problem, *Image Vis. Comput.* 30 (3) (2012) 177–185.
- [12] B.F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, A.K. Jain, Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus benchmark A, in: *Proceedings of the CVPR*, 2015, pp. 1931–1939.
- [13] M. Wang, W. Deng, Deep visual domain adaptation: a survey, *Neurocomputing* 312 (2018) 135–153, doi:10.1016/j.neucom.2018.05.083.
- [14] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, T. Darrell, Deep domain confusion: maximizing for domain invariance, *Comput. Sci.* (2014) arXiv preprint arXiv:1412.3474.
- [15] M. Long, Y. Cao, J. Wang, M.I. Jordan, Learning transferable features with deep adaptation networks, in: *Proceedings of the International Conference on International Conference on Machine Learning*, 2015, pp. 97–105.
- [16] M. Long, J. Wang, M.I. Jordan, Deep Transfer Learning With Joint Adaptation Networks, arXiv:1605.06636 (2016).
- [17] M. Long, H. Zhu, J. Wang, M.I. Jordan, Unsupervised domain adaptation with residual transfer networks, in: *Proceedings of the Advances in Neural Information Processing Systems*, 2016, pp. 136–144.
- [18] N. Courty, R. Flamary, D. Tuia, A. Rakotomamonjy, Optimal transport for domain adaptation, *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (9) (2017) 1853–1865.
- [19] N. Courty, R. Flamary, A. Habrard, A. Rakotomamonjy, Joint distribution optimal transportation for domain adaptation, in: *Proceedings of the Advances in Neural Information Processing Systems*, 2017, pp. 3730–3739.
- [20] W. Zellinger, T. Grubinger, E. Lughofer, T. Natschlager, S. Saminger-Platz, Central Moment Discrepancy (CMD) for Domain-Invariant Representation Learning, arXiv:1702.08811 (2017).
- [21] E. Tzeng, J. Hoffman, K. Saenko, T. Darrell, Adversarial Discriminative Domain Adaptation, arXiv:1702.05464 (2017).
- [22] Y. Ganin, V. Lempitsky, Unsupervised domain adaptation by backpropagation, in: *Proceedings of the International Conference on Machine Learning*, 2015, pp. 1180–1189.
- [23] Z. Pei, Z. Cao, M. Long, J. Wang, Multi-adversarial domain adaptation, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.
- [24] W. Zhang, W. Ouyang, W. Li, D. Xu, Collaborative and adversarial network for unsupervised domain adaptation, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3801–3809.
- [25] E. Tzeng, J. Hoffman, T. Darrell, K. Saenko, Simultaneous deep transfer across domains and tasks, in: *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4068–4076.
- [26] S. Motiian, Q. Jones, S. Iranmanesh, G. Doretto, Few-shot adversarial domain adaptation, in: *Proceedings of the Advances in Neural Information Processing Systems*, 2017, pp. 6670–6680.
- [27] M. Ghifary, W.B. Kleijn, M. Zhang, D. Balduzzi, W. Li, Deep reconstruction-classification networks for unsupervised domain adaptation, in: *Proceedings of the European Conference on Computer Vision*, Springer, 2016, pp. 597–613.
- [28] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, D. Erhan, Domain separation networks, in: *Proceedings of the Advances in Neural Information Processing Systems*, 2016, pp. 343–351.
- [29] J. Ni, Q. Qiu, R. Chellappa, Subspace interpolation via dictionary learning for unsupervised domain adaptation, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 692–699.
- [30] M. Kan, J. Wu, S. Shan, X. Chen, Domain adaptation for face recognition: targetize source domain bridged by common subspace, *Int. J. Comput. Vis.* 109 (1–2) (2014) 94–109.
- [31] K. Saito, Y. Ushiku, T. Harada, Asymmetric Tri-Training for Unsupervised Domain Adaptation, arXiv:1702.08400 (2017).
- [32] C. Chen, W. Xie, T. Xu, W. Huang, Y. Rong, X. Ding, Y. Huang, J. Huang, Progressive Feature Alignment for Unsupervised Domain Adaptation, arXiv:1811.08585 (2018).
- [33] M. Chen, K.Q. Weinberger, J. Blitzer, Co-training for domain adaptation, in: *Proceedings of the Advances in Neural Information Processing Systems*, 2011, pp. 2456–2464.
- [34] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A.K. Jain, J.A. Duncan, K. Allen, et al., Iarpa janus benchmark-b face dataset, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 90–98.
- [35] B. Maze, J. Adams, J.A. Duncan, N. Kalka, T. Miller, C. Otto, A.K. Jain, W.T. Niggel, J. Anderson, J. Cheney, et al., IARPA Janus benchmark-c: face dataset and proto-

- col, in: Proceedings of the International Conference on Biometrics (ICB), IEEE, 2018, pp. 158–165.
- [36] M. Wang, W. Deng, J. Hu, X. Tao, Y. Huang, Racial faces in the wild: reducing racial bias by information maximization adaptation network, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 692–702.
- [37] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in: Proceedings of the CVPR, 2014, pp. 1701–1708.
- [38] M. Wang, W. Deng, Deep Face Recognition: A Survey, arXiv:1804.06655 (2018).
- [39] W.-S.T. WST, Deeply learned face representations are sparse, selective, and robust, *Perception* 31 (2008) 411–438.
- [40] Y. Sun, D. Liang, X. Wang, X. Tang, 2015. arXiv: 1502.00873
- [41] Y. Sun, Y. Chen, X. Wang, X. Tang, Large-margin softmax loss for convolutional neural networks, in: Proceedings of the NIPS, 2014, pp. 1988–1996.
- [42] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the CVPR, 2015, pp. 815–823.
- [43] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: Proceedings of the ECCV, Springer, 2016, pp. 499–515.
- [44] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, L. Song, Sphreface: deep hypersphere embedding for face recognition, in: Proceedings of the CVPR, 1, 2017.
- [45] W. Liu, Y. Wen, Z. Yu, M. Yang, Large-margin softmax loss for convolutional neural networks, in: Proceedings of the International Conference on International Conference on Machine Learning, 2016, pp. 507–516.
- [46] H. Wang, Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, W. Liu, Cosface: Large Margin Cosine Loss For Deep Face Recognition, arXiv:1801.09414 (2018).
- [47] F. Wang, J. Cheng, W. Liu, H. Liu, Additive margin softmax for face verification, *IEEE Signal Process. Lett.* 25 (7) (2018) 926–930.
- [48] J. Deng, J. Guo, S. Zafeiriou, 2018. arXiv: 1801.07698
- [49] R. Ranjan, C.D. Castillo, R. Chellappa, L2-Constrained Softmax Loss for Discriminative Face Verification, arXiv:1703.09507 (2017).
- [50] Y. Zheng, D.K. Pal, M. Savvides, Ring loss: convex feature normalization for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5089–5097.
- [51] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, W. Zuo, Mind the Class Weight Bias: Weighted Maximum Mean Discrepancy for Unsupervised Domain Adaptation, arXiv:1705.00609 (2017).
- [52] S. Xie, Z. Zheng, L. Chen, C. Chen, Learning semantic representations for unsupervised domain adaptation, in: Proceedings of the International Conference on Machine Learning, 2018, pp. 5419–5428.
- [53] Z.-H. Zhou, M. Li, Tri-training: exploiting unlabeled data using three classifiers, *IEEE Trans. Knowl. Data Eng.* 17 (11) (2005) 1529–1541.
- [54] Z. Cao, M. Long, J. Wang, M.I. Jordan, Partial transfer learning with selective adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2724–2732.
- [55] J. Zhang, Z. Ding, W. Li, P. Ogunbona, Importance weighted adversarial nets for partial domain adaptation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8156–8164.
- [56] P. Panareda Busto, J. Gall, Open set domain adaptation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 754–763.
- [57] H. Liu, Z. Cao, M. Long, J. Wang, Q. Yang, Separate to adapt: Open set domain adaptation via progressive separation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2927–2936.
- [58] K. Saito, S. Yamamoto, Y. Ushiku, T. Harada, Open set domain adaptation by backpropagation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 153–168.
- [59] B. Yang, A.J. Ma, P.C. Yuen, Learning domain-shared group-sparse representation for unsupervised domain adaptation, *Pattern Recognit.* 81 (2018) 615–632.
- [60] M. Gheisari, M.S. Baghshah, Unsupervised domain adaptation via representation learning and adaptive classifier learning, *Neurocomputing* 165 (2015) 300–311.
- [61] J. Tao, W. Hu, S. Wang, Sparsity regularization label propagation for domain adaptation learning, *Neurocomputing* 139 (2014) 202–219.
- [62] Y. Zong, W. Zheng, X. Huang, J. Shi, Z. Cui, G. Zhao, Domain regeneration for cross-database micro-expression recognition, *IEEE Trans. Image Process.* 27 (5) (2018) 2484–2498.
- [63] W.D.H.S. Zimeng Luo Jiani Hu, Deep unsupervised domain adaptation for face recognition, in: Proceedings of the FG, IEEE, 2018, pp. 453–457.
- [64] K. Sohn, S. Liu, G. Zhong, X. Yu, M.-H. Yang, M. Chandraker, Unsupervised Domain Adaptation for Face Recognition in Unlabeled Videos, arXiv:1708.02191 (2017).
- [65] M. Kan, S. Shan, X. Chen, Bi-shifting auto-encoder for unsupervised domain adaptation, in: Proceedings of the ICCV, 2015, pp. 3846–3854.
- [66] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf, A. Smola, A kernel two-sample test, *J. Mach. Learn. Res.* 13 (Mar) (2012) 723–773.
- [67] D.-H. Lee, Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks, in: Proceedings of the Workshop on Challenges in Representation Learning, ICML, 3, 2013, p. 2.
- [68] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks? in: Proceedings of the Advances in Neural Information Processing Systems, 2014, pp. 3320–3328.
- [69] P.J. Phillips, P.J. Flynn, T. Scruggs, K.W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, W. Worek, Overview of the face recognition grand challenge, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005., 1, IEEE, 2005, pp. 947–954.
- [70] A. Gretton, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, K. Fukumizu, B.K. Sriperumbudur, Optimal kernel choice for large-scale two-sample tests, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1205–1213.
- [71] A.R. Chowdhury, T.-Y. Lin, S. Maji, E. Learned-Miller, One-to-many face recognition with bilinear CNNs, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016, pp. 1–9.
- [72] D. Wang, C. Otto, A.K. Jain, Face Search at Scale: 80 Million Gallery, arXiv:1507.07242 (2015).
- [73] W. AbdAlmageed, Y. Wu, S. Rawls, S. Harel, T. Hassner, I. Masi, J. Choi, J. Lekust, J. Kim, P. Natarajan, et al., Face recognition using deep multi-pose representations, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016, pp. 1–9.
- [74] S. Sankaranarayanan, A. Alavi, R. Chellappa, Triplet Similarity Embedding for Face Verification, arXiv:1602.03418 (2016).
- [75] J.-C. Chen, V.M. Patel, R. Chellappa, Unconstrained face verification using deep CNN features, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016, pp. 1–9.
- [76] N. Bodla, J. Zheng, H. Xu, J.-C. Chen, C. Castillo, R. Chellappa, Deep heterogeneous feature fusion for template-based face recognition, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2017, pp. 586–595.
- [77] L. Tran, X. Yin, X. Liu, Disentangled representation learning Gan for pose-invariant face recognition, in: Proceedings of the CVPR, 3, 2017, p. 7.
- [78] B. Yin, L. Tran, H. Li, X. Shen, X. Liu, Towards interpretable face recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 9348–9357.
- [79] M. Wang, W. Deng, Mitigate Bias in Face Recognition Using Skewness-Aware Reinforcement Learning, arXiv:1911.10692 (2019).
- [80] L.v.d. Maaten, G. Hinton, Visualizing data using T-SNE, *J. Mach. Learn. Res.* 9 (Nov) (2008) 2579–2605.



Mei Wang received the B.E. degree in information and communication engineering from the Dalian University of Technology (DUT), Dalian, China, in 2013 and received M.E. degree in communication engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2016. From September 2018, she is a Ph.D. student in school of information and communication engineering of BUPT. Her research interests include pattern recognition and computer vision, with a particular emphasis in deep face recognition and transfer learning.



Weihong Deng received the B.E. degree in information engineering and the Ph.D. degree in signal and information processing from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2004 and 2009, respectively. From Oct. 2007 to Dec. 2008, he was a postgraduate exchange student in the School of Information Technologies, University of Sydney, Australia. He is currently an professor in School of Information and Telecommunications Engineering, BUPT. His research interests include statistical pattern recognition and computer vision, with a particular emphasis in face recognition. He has published over 100 technical papers in international journals and conferences, such as IEEE TPAMI and CVPR. He serves as associate editor for IEEE Access, and guest editor for Image and Vision Computing Journal and the reviewer for dozens of international journals, such as IEEE TPAMI / TIP / TIFS / TNNLS / TMM / TSMC, IJCV, PR / PRL. His Dissertation titled “Highly accurate face recognition algorithms” was awarded the Outstanding Doctoral Dissertation Award by Beijing Municipal Commission of Education in 2011. He has been supported by the program for New Century Excellent Talents by the Ministry of Education of China in 2013 and Beijing Nova Program in 2016.