Deep Compact Discriminative Representation for Unconstrained Face Recognition

Monica M.Y. Zhang[a], Kun Shang[b,*], Huaming Wu[c]

[a] *College of Information Science and Engineering, Henan University of Technology, Zhengzhou, 450001, PR China.*
[b] *College of Mathematics and Econometrics, Hunan University, Changsha, 410082, PR China.*
[c] *Center for Applied Mathematics, Tianjin University, Tianjin, 300072, PR China.*

Abstract

Convolutional Neural Network has been widely used in pattern recognition community, especially face recognition. Loss function, as a supervisory signal to learn a CNN model, plays an important role in obtaining the desired facial features. However, how to design a loss function to make the features more compact and discriminative for unconstrained face recognition, is still an open problem. In this paper, we propose two novel loss functions, Compact Discriminative loss and Advanced Compact Discriminative loss. They supervise CNN to map the raw data onto the face feature space, where the intra-class space is compact and inter-class spaces have sensible gaps, by constraining the intra-class variations and the inter-class variations simultaneously. Three CNNs (i.e. LeNet, CNN-M and ResNet-50) are used to analyze the effectiveness of the proposed approaches, the obtained models are evaluated on several famous benchmark databases, such as MNIST, LFW, FGLFW, YTF and IJB-A. Experimental results show that the proposed losses are effective for face recognition, and can easily generate comparable results than related state-of-the-art methods.

Keywords: Convolutional neural network, Compact discriminative loss, Advanced compact discriminative loss, Deep compact discriminative representation, Face recognition.

*Corresponding author.
Email address: skun@hnu.edu.cn (Kun Shang)

1. Introduction

Face recognition, non-intrusive and natural, has been widely studied in computer vision and pattern recognition community due to its close relationship with many real-world applications, such as human-machine interaction, digital entertainment, photo album management in social networks and commercial security system. Both face verification task and face identification task contain two main stages: one is feature extraction (e.g. Local Binary Patterns [1], Gabor [2] and Scale-Invariant Feature Transform [3]), and the other one is classification (e.g. Nearest Neighbor [4], K-Nearest Neighbor [5], Sparse Representation Classification [6], Collaborative Representation Classification [7], and their variants [8, 9]). Particularly, feature extraction plays an important role because the representation capacity of the feature influences the performance of face recognition. With suitable classifiers, the mentioned feature extraction methods have achieved respectable performance on many face recognition tasks. However, the representations composed by hand-crafted descriptors are too shallow to satisfy the increasing demand for more and more complex applications. Especially, when it comes to unconstrained environments, the performance may degrade dramatically due to the complex and large intra-personal variations, such as pose, illumination and occlusion.

Nowadays, Convolutional Neural Network (CNN), which emerging as an automatic and powerful feature extraction method, has achieved impressive results [10, 11, 12, 13, 14]. Notably, the methods based on CNN [15, 16, 17, 18, 19] continuously won the champions of the ImageNet LSVRC contests[1] from 2012 to present. These phenomenal successes make more and more researchers pay attention to the development of CNN. It should be stressed here that deep CNNs, such as DeepID series [20, 21, 22, 23], DeepFace [24], FaceNet [25], VGG [26], ResNet based approach [27, 28], have achieved great success on face

[1] [Online]. Available: `http://www.image-net.org/challenges/LSVRC/`

recognition. These approaches even made the face recognition systems surpass human-level face verification performance on LFW database [29]. CNN based feature extraction becomes a new trend for face recognition.

For enhancing the performance of CNN based feature extraction on face recognition, a variety of methods have been proposed in recent years, which can be mainly classified into four categories:

- producing powerful CNN architecture [21, 26, 25, 30];
- pursuing high quality data preprocessing [24, 31];
- extending the training face images to million orders of magnitude [26, 25, 31];
- designing suitable loss functions [21, 25, 32, 28, 33, 34, 35].

To our best knowledge, designing suitable loss functions, that supervising CNNs to obtain face features with high discrimination, is one of the most simple and effective efforts to improve performance for face recognition. The typical method can even achieve comparable face verification performance with less than 0.5 million training face images compared to the others. However, how to design a loss function to make the learned features more suitable for face recognition, even more compact and discriminative in unconstrained circumstances, is still an open problem.

In this paper, we focus on constructing suitable loss functions to supervise CNN for more compact and discriminative face representations. To this end, two loss functions, Compact Discriminative (CD) loss and Advanced Compact Discriminative (ACD) loss, are proposed. Both the losses supervise CNN to map the raw data onto the feature space, where the intra-class space is compact and inter-class spaces have sensible gaps, by adaptively constraining the intra-class variations and the inter-class variations. Further, ACD loss is designed to alleviate the imbalanced computation of CD loss. To illustrate the effectiveness and the adaption of our proposed loss functions, we conduct extensive experiments for face recognition, which consist of a small-scale face verification

task, three different levels of large-scale image-to-image face verification tasks, a large-scale video-to-video face verification task, a template-to-template face verification task, and two challenging face identification tasks. Famous public benchmark databases, including MNIST [10], LFW [29, 36], FGLFW [37] and YTF [38] are used for evaluation. Experimental results show that our proposed loss functions are effective, and can easily generate more comparable results with some existing state-of-the-art methods.

The remainder of this paper is organized as follows: Section 2 illustrates the related works; Section 3 describes the proposed approaches; Section 4 provides a wide range of experiments; Section 5 gives the conclusion.

2. Related works

As is shown in Fig. 1, CNN based feature extraction benefits from three primary attributes: the available training data, the suitable CNN architecture, and the carefully designed loss function. Here we focus on the loss function to enhance the discriminative power of the deeply learned face features.

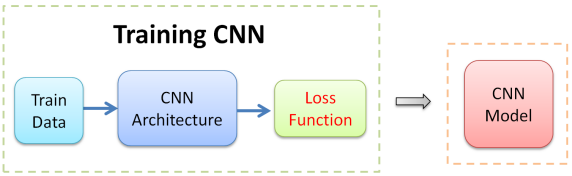

Fig. 1: The figure shows a representative framework about obtaining a desired CNN model for extracting face features. The loss function, one of the three primary attributes, plays an important role for learning the CNN model.

Generally, the optimization objective can be expressed as

$$\boldsymbol{\theta}^* = \min_{\boldsymbol{\theta}} \mathcal{L}(X, R, \boldsymbol{\theta})$$

where $\mathcal{L}(X, R, \boldsymbol{\theta})$ is a general loss function to supervise the CNN model, $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ is the training data set, $R = \{r_1, r_2, \ldots, r_n\}$ is the corresponding real label set, and $\boldsymbol{\theta}$ is the parameter set. In the proposed approaches,

we also use the predicted label p_m $(m = 1, \ldots, n)$, which is obtained from the softmax prediction in the last layer of a given CNN.

For most CNN models, Softmax loss $\mathcal{L}_S$ is the favorite supervisory signal:

$$\mathcal{L}_S = -\frac{1}{M}\sum_{m=1}^{M}\log\frac{e^{\boldsymbol{w}_{r_m}^{\top}\hat{\boldsymbol{x}}_m+\boldsymbol{b}_{r_m}}}{\sum_{j=1}^{N}e^{\boldsymbol{w}_j^{\top}\hat{\boldsymbol{x}}_m+\boldsymbol{b}_j}},$$

where $\hat{\boldsymbol{x}}_m$ is the CNN feature for the m-th sample $\boldsymbol{x}_m$, M is the mini-batch size, N is the class number, $\boldsymbol{w}_j$ and $\boldsymbol{b}_j$ are the parameters that belong to $\boldsymbol{\theta}$. Softmax loss is a classical and effective loss for supervising CNN to obtain face features. Unfortunately, the learned features still lack sufficient discriminative information.

To enhance the discrimination of the face features, Schroff *et al.* [25] proposed Triplet loss $\mathcal{L}_T$:

$$\mathcal{L}_T = \sum_{m=1}^{M}[\|f(\hat{\boldsymbol{x}}_m^a) - f(\hat{\boldsymbol{x}}_m^p)\|^2 - \|f(\hat{\boldsymbol{x}}_m^a) - f(\hat{\boldsymbol{x}}_m^n)\|^2 + \phi]_+,$$

where ϕ is a relative distance constraint, $(\hat{\boldsymbol{x}}_m^a, \hat{\boldsymbol{x}}_m^p, \hat{\boldsymbol{x}}_m^n)$ is a triplet, $\hat{\boldsymbol{x}}_m^a$ and $\hat{\boldsymbol{x}}_m^p$ are in the same class, $\hat{\boldsymbol{x}}_m^a$ and $\hat{\boldsymbol{x}}_m^n$ are in the different classes. It makes a triplet constraint for reducing the intra-class variations and enlarging the inter-class variations, simultaneously. However, the selection of triplets is not an easy work, which may result in a collapsed model.

Recently, Wen *et al.* [28] proposed Center loss $\mathcal{L}_c$ to assist Softmax loss $\mathcal{L}_S$ with the identity-related information instead of triplet information to enhance the discrimination of the face features. The whole loss function is formalized as

$$\mathcal{L}_C = \mathcal{L}_S + \lambda\mathcal{L}_c,$$

where

$$\mathcal{L}_c = \frac{1}{2M}\sum_{m=1}^{M}\|\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{r_m}\|^2,$$

$\boldsymbol{c}_{r_m}$ denotes the center feature for r_m-th class. $\mathcal{L}_c$ is designed to enhance the discriminative power of the features by penalizing the distances between the features and their corresponding center features.

Other famous approaches to enhance the discrimination of CNN learned face features include NormFace [33], L-Softmax [27] and SphereFace [32]. NormFace generalized the center feature to an concept of "agent vector" for each class, they studied the effect of normalization during training and optimized cosine similarity instead of inner-product, finally improved performance by between 0.2% to 0.4% on LFW by finetuning the CNN model released in [33]. L-Softmax employed a margin constraint in the original Softmax loss. SphereFace extended L-Softmax loss by considering the cosine normalization, and achieved excellent performance on face recognition by adopting deeper well designed CNN architectures.

Review the loss functions mentioned above, Softmax loss only considers the identity-related information for separating features into different classes. Triplet loss needs a lot of manual intervention to deal with the intractable selection of the training triplets for increasing the inter-class variations and reducing the intra-class variations. Center loss focuses on the relationship with the feature and the center feature in the same class while does not emphasize the relationship with the inter-class features. L-Softmax may face more difficult convergence problem than Softmax loss when there are too many subjects. NormFace and SphereFace depend on careful designed CNN architectures and need to optimize cosine similarity instead of trivial inner-product. These phenomena motivate us to find an easy and adaptive way to further improve the performance of the learned features.

3. Proposed approaches

Given the CNN details, and treating it as a black box (see Fig. 1), the most important part of our approach lies in extracting the compact and discriminative features in the end-to-end learning. To this end, we propose Compact Discriminative (CD) loss for creating a feature space where the intra-class features are as close as possible and the inter-class features are as far as possible. And then, Advanced Compact Discriminative (ACD) loss is proposed to allevi-

ate the imbalanced computation of CD loss. Both the proposed loss functions reduce the differences of the intra-class features and make the inter-class features to have sensible gaps corresponding to the comparison of the real label and the predicted label, which is obtained from the softmax prediction in the last layer of a given CNN.

For convenience, we use the following unified notations in Table 1 and only consider the computation in a mini-batch without declaration.

Table 1: Unified notations.

Notations	
M	the mini-batch size
N	the class number
$\boldsymbol{x}_m$	the m-th training sample
$\hat{\boldsymbol{x}}_m$	the CNN feature of $\boldsymbol{x}_m$
$\boldsymbol{c}_m$	the center feature for m-th class
r_m	the real label for $\boldsymbol{x}_m$ or $\hat{\boldsymbol{x}}_m$
p_m	the predicted label for $\boldsymbol{x}_m$ or $\hat{\boldsymbol{x}}_m$

3.1. Compact discriminative loss

To give the concept of Compact Discriminative loss, we first introduce two functions, namely, intra compact function and inter discriminative function, which characterize the relationship with intra-class features or the relationship with the inter-class features, respectively.

3.1.1. Intra compact function

Intra compact function F_{IC} measures the distance[2] between the feature $\hat{\boldsymbol{x}}_m$ and the corresponding center feature $\boldsymbol{c}_{p_m}$, which is defined as

$$F_{IC} = \mathbb{I}(p_m = r_m)\|\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m}\|^2,$$

[2]We simply adopt the commonly used L2-distance, other distances are beyond the scope of consideration.

where $\mathbb{I}$ is an indicator function, defined by

$$\mathbb{I}(\text{condition}) = \begin{cases} 1 & \text{if the condition is true,} \\ 0 & \text{otherwise.} \end{cases}$$

It aims to make sure that the intra-class features of a specific class are close to the corresponding center feature, which causes the intra-class space to be compact.

3.1.2. Inter discriminative function

In contrast, inter discriminative function F_{ID} focuses on the distance between the misclassified feature $\hat{\boldsymbol{x}}_m$ ($p_m \neq r_m$) and the feature $\hat{\boldsymbol{x}}_t$ in p_m-th class : $\|\hat{\boldsymbol{x}}_m - \hat{\boldsymbol{x}}_t\|^2$, it is defined as

$$F_{ID} = \sum_{t=1}^{T_m} \mathbb{I}(p_m \neq r_m) \|\hat{\boldsymbol{x}}_m - \hat{\boldsymbol{x}}_t\|^2,$$

where T_m is the number of features in p_m-th class. It aims to force the misclassified features $\hat{\boldsymbol{x}}_m$ be away from its predicted (p_m-th) class for expecting $\hat{\boldsymbol{x}}_m$ to return to its true (r_m-th) class.

Combining the above two defined functions in a mini-batch M, Compact Discriminative (CD) loss $\mathcal{L}_{cd}$ is defined as

$$\begin{aligned} \mathcal{L}_{cd} &= \frac{1}{2M} \sum_{m=1}^{M} [\tau F_{IC} - (1-\tau) F_{ID}] \\ &= \frac{1}{2M} \sum_{m=1}^{M} [\tau \mathbb{I}(p_m = r_m) \|\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m}\|^2 \\ &\quad -(1-\tau) \sum_{t=1}^{T_m} \mathbb{I}(p_m \neq r_m) \|\hat{\boldsymbol{x}}_m - \hat{\boldsymbol{x}}_t\|^2], \end{aligned}$$

where $\tau \in (0, 1)$ is a hyper-parameter to balance F_{IC} and F_{ID}. By minimizing $\mathcal{L}_{cd}$ when fixing a m, the gap between the feature $\hat{\boldsymbol{x}}_m$ and each feature $\hat{\boldsymbol{x}}_t$ in the p_m-th class will be widened if $p_m \neq r_m$, and the distance between $\hat{\boldsymbol{x}}_m$ and the corresponding center feature $\boldsymbol{c}_{p_m}$ will be penalized if $p_m = r_m$. Considering all features in a mini-batch ($m \in \{1, 2, \ldots, M\}$), $\mathcal{L}_{cd}$ will give clear restraint

for both the inter-class variations and the intra-class variations during the CNN optimization.

By taking advantage of the joint supervision of Softmax loss and CD loss for CNN, the whole loss function is formalized as

$$\mathcal{L}_{CD} = \mathcal{L}_S + \lambda \mathcal{L}_{cd}, \tag{1}$$

where $\lambda \in (0, 1)$ is a trade-off hyper-parameter. The flowchart about the joint supervision is shown in Fig. 2.

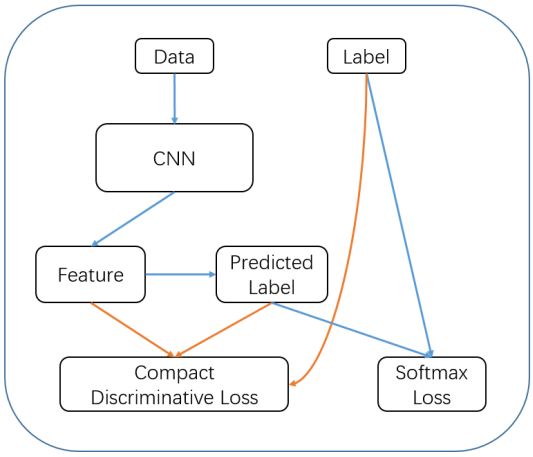

Fig. 2: A flowchart about joint supervision of Softmax loss and CD loss for CNN. Particularly, CD loss supervises the learning of CNN by judging whether a feature is misclassified, where the judge reflects in the feature's attribute (the comparison of the real label and the predicted label).

3.2. Advanced compact discriminative loss

Aforementioned, CD loss $\mathcal{L}_{cd}$ aims to enlarge the gaps between the inter-class spaces by reducing the sum of distances between the misclassified feature $\hat{\boldsymbol{x}}_m$ $(p_m \neq r_m)$ and the features in the p_m-th class. Although it avoids using plenty of triplet features, it still faces the imbalanced problem in computation. Specifically, it computes T_m (the number of features in p_m-th class) times inter-distances for $\hat{\boldsymbol{x}}_m$ $(p_m \neq r_m)$, while only computes once intra-distance.

To alleviate this problem, we propose Advanced Compact Discriminative (ACD) loss, which is defined by

$$\mathcal{L}_{acd} = \frac{1}{2M}\sum_{m=1}^{M}[\tau F_{IC} - (1-\tau)F_{AID}],$$

where

$$F_{AID} = \mathbb{I}(p_m \neq r_m)\|\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m}\|^2.$$

It alleviates the imbalanced computation by only computing the distance between $\hat{\boldsymbol{x}}_m$ and the center feature $\boldsymbol{c}_{p_m}$ for inter-distance. Similarly, the whole loss supervising CNN is formalized as

$$\mathcal{L}_{ACD} = \mathcal{L}_S + \lambda\mathcal{L}_{acd}. \tag{2}$$

3.3. Computation and algorithm

In the subsection, we tell how the proposed losses supervise the feature learning for CNN. The optimization objectives corresponding to the proposed approaches are

$$\boldsymbol{\theta}_1^* = \min_{\boldsymbol{\theta}} \mathcal{L}_{CD}(X, R, \boldsymbol{\theta}), \tag{3}$$

$$\boldsymbol{\theta}_2^* = \min_{\boldsymbol{\theta}} \mathcal{L}_{ACD}(X, R, \boldsymbol{\theta}), \tag{4}$$

which can be easily optimized by the stochastic gradient descent algorithm.

For (3), the gradients of $\mathcal{L}_{cd}$ with respect to $\hat{\boldsymbol{x}}_m$ and $\boldsymbol{c}_{p_m}$ are

$$\begin{aligned}\frac{\partial\mathcal{L}_{cd}}{\partial\hat{\boldsymbol{x}}_m} &= \frac{1}{2M}\sum_{m=1}^{M}[\tau\frac{\partial F_{IC}}{\partial\hat{\boldsymbol{x}}_m} - (1-\tau)\frac{\partial F_{ID}}{\partial\hat{\boldsymbol{x}}_m}] \\ &= \frac{1}{M}\sum_{m=1}^{M}[\tau\mathbb{I}(p_m = r_m)(\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m}) \\ &\quad -(1-\tau)\sum_{t=1}^{T_m}\mathbb{I}(p_m \neq r_m)(\hat{\boldsymbol{x}}_m - \hat{\boldsymbol{x}}_t)],\end{aligned} \tag{5}$$

and

$$\begin{aligned}\frac{\partial\mathcal{L}_{cd}}{\partial\boldsymbol{c}_{p_m}} &= \frac{\tau}{2M}\sum_{m=1}^{M}\frac{\partial F_{IC}}{\partial\boldsymbol{c}_{p_m}} \\ &= \frac{\tau}{M}\sum_{m=1}^{M}\mathbb{I}(p_m = r_m)(\boldsymbol{c}_{p_m} - \hat{\boldsymbol{x}}_m).\end{aligned} \tag{6}$$

Similarly, for (4), the gradients of $\mathcal{L}_{acd}$ with respect to $\hat{\boldsymbol{x}}_m$ and $\boldsymbol{c}_{p_m}$ are

$$\begin{aligned}\frac{\partial \mathcal{L}_{acd}}{\partial \hat{\boldsymbol{x}}_m} &= \frac{1}{2M}\sum_{m=1}^{M}[\tau\frac{\partial F_{IC}}{\partial \hat{\boldsymbol{x}}_m} - (1-\tau)\frac{\partial F_{AID}}{\partial \hat{\boldsymbol{x}}_m}] \\ &= \frac{1}{M}\sum_{m=1}^{M}[\tau\mathbb{I}(p_m = r_m)(\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m}) \\ &\quad -(1-\tau)\mathbb{I}(p_m \neq r_m)(\hat{\boldsymbol{x}}_m - \boldsymbol{c}_{p_m})], \end{aligned} \tag{7}$$

and

$$\begin{aligned}\frac{\partial \mathcal{L}_{acd}}{\partial \boldsymbol{c}_{p_m}} &= \frac{1}{2M}\sum_{m=1}^{M}[\tau\frac{\partial F_{IC}}{\partial \boldsymbol{c}_{p_m}} - (1-\tau)\frac{\partial F_{AID}}{\partial \boldsymbol{c}_{p_m}}] \\ &= \frac{1}{M}\sum_{m=1}^{M}[\tau\mathbb{I}(p_m = r_m)(\boldsymbol{c}_{p_m} - \hat{\boldsymbol{x}}_m) \\ &\quad -(1-\tau)\mathbb{I}(p_m \neq r_m)(\boldsymbol{c}_{p_m} - \hat{\boldsymbol{x}}_m)]. \end{aligned} \tag{8}$$

Here we only take Compact Discriminative loss for example, the corresponding algorithm is summarized in **Algorithm 1**.

Algorithm 1 Deep compact discriminative representation learning algorithm

Input: Training data set $\{X, R\} = \{(\boldsymbol{x}_1, r_1), \ldots, (\boldsymbol{x}_n, r_n)\}$, two hyper-parameters λ and τ, center learning rate γ; mini-batch size M, maximum iteration t_{max}, weight decay μ, momentum α, learning policy κ, step size set ς, learning rate β; center feature $\boldsymbol{c}_{p_m}^t$, parameters $\boldsymbol{\theta}^t$ and $\boldsymbol{\delta}^t$, $t \leftarrow 0$.

Output: Parameters $\boldsymbol{\theta}^{t_{max}}$;

while not convergence and $t < t_{max}$ **do:**

1. $t = t + 1$;

2. Compute CD loss by (1);

3. Update the centers $\boldsymbol{c}_{p_m}^{t+1} = \boldsymbol{c}_{p_m}^t - \gamma\frac{\partial \mathcal{L}_{cd}}{\partial \boldsymbol{c}_{p_m}^t}$ by (6);

4. Update the parameters $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \boldsymbol{\delta}^{t+1}$ by (5), where $\boldsymbol{\delta}^{t+1} = \alpha\boldsymbol{\delta}^t - \beta[\sum_m(\frac{\partial \mathcal{L}_S}{\partial \hat{\boldsymbol{x}}_m^t} + \lambda\frac{\partial \mathcal{L}_{cd}}{\partial \hat{\boldsymbol{x}}_m^t})\frac{\partial \hat{\boldsymbol{x}}_m^t}{\partial \boldsymbol{\theta}^t} + \mu\boldsymbol{\theta}^t]$;

if t is divisible by $\hat{\varsigma} \in \varsigma$ **do:**

$$\beta = \kappa\beta;$$

end if

end while

3.4. Discussions

The strategy, combining Softmax loss $\mathcal{L}_S$ and an auxiliary loss ($\mathcal{L}_c$, $\mathcal{L}_{cd}$ or $\mathcal{L}_{acd}$) to jointly supervise CNN, reinforces the learned features with more discriminative information than only using Softmax loss $\mathcal{L}_S$. In addition, this strategy, which avoiding the inescapable step of selecting training triplets in $\mathcal{L}_T$, is more easy and trainable. Although $\mathcal{L}_c$ appears as a simple and effective way to learn discriminative features, it mainly focuses on the intra-class relationship, which may give not enough treatment for the inter-class variations. Intuitively, giving clear statement for both intra-class relationship and the inter-class relationship can further restrain the intra-class variations and the inter-class variations for more accurate performance.

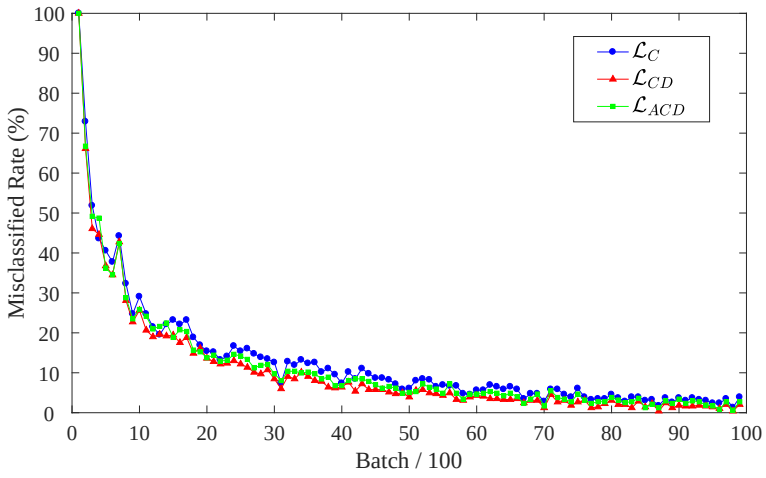

Fig. 3: Trained on LeNet with MNIST database, the misclassified rate is displayed for $\mathcal{L}_C$, $\mathcal{L}_{CD}$ and $\mathcal{L}_{ACD}$, respectively.

Either $\mathcal{L}_{cd}$ or $\mathcal{L}_{acd}$ compresses the intra-class feature space and enlarges the gaps between inter-class feature spaces simultaneously to supervise the learning of CNN by judging whether a feature is misclassified or not. In fact, the experiment results, as shown in Fig. 3, demonstrates that the proposed losses exactly impose better restraint for the misclassified rate (the normalization of the total number of the misclassified features every 100 batches), compared to

$\mathcal{L}_C$. More importantly, compared to $\mathcal{L}_{cd}$, $\mathcal{L}_{acd}$ takes advantage of both the intra-class information and inter-class information to update the center features in (8) compared to $\mathcal{L}_{cd}$'s intra-class information in (6), which seems more beneficial to improve the discrimination of the learned feature.

4. Experiments

In this section, we use three CNN architectures and six databases to demonstrate the effectiveness of the proposed CD loss and ACD loss on face verification tasks and face identification tasks. All experiments are implemented in the Caffe library [40] on Linux OS with the NVIDIA Tesla K80.

4.1. Experimental setup

4.1.1. Databases

- MNIST [10] is a classical handwritten digit database, which contains 60,000 training examples and 10,000 testing examples. It is used to illustrate the effectiveness of the proposed approaches for restricting the misclassified rate in Fig. 3.

- LFW [29] is one of the most challenging face databases, which has been widely used for image-to-image face recognition. We choose it as a benchmark database for face verification in Subsection 4.3 and for face identification in Subsection 4.4. Besides, we use LFW-SUB, which is the verification subset of LFW and consisted of 10 splits of face matches, for small-scale face verification in Subsection 4.2. And the related details are as same as the descriptions in the literature [41, 39, 42].

- FGLFW [37] is a database which shares the same $3,000$ genius matches in LFW, however, replaces the random impostor matches by seeking another $3,000$ similarly-looking face pairs to reduce the inter-class variance. It emphasizes both the large intra-class variance and the tiny inter-class variance simultaneously compared to LFW. We choose it as a more challenging image-to-image face verification benchmark in Subsection 4.3.

- YTF database [38] is a popular video-to-video face verification benchmark database, which contains $3,425$ videos of $1,595$ different people. Each subject contains several videos with different size of frames ranging from 48 to $6,070$. It not only has large unconstrained face variations, but also suffers from different levels of low resolutions. The test set of YTF contains $5,000$ pairs of face videos, dividing into 10 splits for reporting performance as LFW does. We choose it as the video-to-video face verification benchmark, shown in Subsection 4.3.

- IJB-A [43] is a public database with full pose variations which contains 500 subjects with manually localized face images of a mixture of images and videos, where every subject in the database contains at least five images and one video. The images in the IJB-A dataset contain extreme pose, illumination and expression variations, which makes it a more challenging mixture face benchmark database for face recognition. We choose it for both template-to-template face verification and template-to-template face identification in Subsection 4.3 and Subsection 4.4, respectively.

- CASIA-WebFace [39] is a typical public face database, which contains $10,575$ subjects and $494,414$ images collected from Internet. We choose it as the training database for obtaining CNN model since it is almost independent of the LFW and YTF benchmarks, and can dispel the chaos of evaluations.

4.1.2. CNN architectures

- LeNet [10] is a famous CNN architecture, which consists 3 convolutional layers, 3 max-pooling layers and 1 fully-connected layer. It is chosen to investigate the influence of the proposed approach toward handwritten digit recognition.

- CNN-M has 3 convolutional layers, 3 max-pooling layers and 1 fully-connected layer, its first appearance was in the literature [41]. We use

it for evaluating the effect of proposed losses on the small-scale face verification on LFW-SUB, which will be described in Subsection 4.2.

- ResNet [19] is a residual learning framework for building deeper network, which introduces a short-cut layer to make the CNN architectures to reach 1000 layers, and has achieved the state-of-the-art performance on many vision tasks. In our experiment, we use the released ResNet-50 architecture and model[3], to investigate the performance of the proposed losses for unconstrained face verification and face identification. For convenience, we modify the ResNet-50 architecture by adding a fully-connected layer of dimension 512.

For LeNet and CNN-M, the output of the last second layer is defined as the CNN feature. For ResNet-50, the CNN feature is defined according to [28].

4.1.3. Parameters

Especially, the parameters in **Algorithm** 1 are divided into two parts: one is the default algorithm parameters (M, t_{max}, μ, α, κ, β and ς), the other one is the parameters related to the proposed losses (τ, γ and λ). The default parameter setting for the related CNNs[4] is according to Table 2.

Table 2: Default parameter setting for each CNN architecture.

CNN	M	t_{max}	μ	α	κ	β	ς
LeNet	64	$10,000$	0.0005	0.9	0.8	0.01	$\{8,000\}$
CNN-M	100	$100,000$	0.0005	0.9	1	0.001	$\{100,000\}$
ResNet-50	28	$20,000$	0.0001	0.9	0.1	0.1	$\{10,000, 15,000\}$

For parameters that related to the proposed losses, we consider the hyper-parameters λ, τ, and the learning rate γ for $\mathcal{L}_{CD}$ and $\mathcal{L}_{ACD}$ on ResNet-50.

[3][Online]. Available: `https://github.com/KaimingHe/deep-residual-networks`

[4]Since ResNet-50 is not so easy to train, we update the parameters of the focused losses every 10 mini-batches when training ResNet-50.

Since the CNN optimization is complex and non-convex, we use cross-validation to find the best hyper-parameters in the premise of CNN convergence, by varying τ in the range $\{0.1, 0.2, \ldots, 0.9\}$, and varying λ and γ in the range $\{0.0001, 0.001, 0.01, 0.1, 1\}$. The final parameter settings are listed in Table 3.

Table 3: Parameter setting for $\mathcal{L}_{CD}$ and $\mathcal{L}_{ACD}$.

	$\mathcal{L}_{CD}$			$\mathcal{L}_{ACD}$		
CNN	τ	γ	λ	τ	γ	λ
LeNet	0.2	0.001	0.001	0.5	0.01	0.001
CNN-M	0.8	0.0001	0.05	0.8	0.0001	0.05
ResNet-50	0.7	0.01	0.01	0.8	0.01	0.01

4.2. Small-scale face verification

Unlike most existing CNN learning based models with large CNN architectures and large private training databases for addressing face verification, we choose CNN-M and LFW-SUB to evaluate whether the proposed loss functions can improve the performance with the small-scale CNN architecture and training database.

4.2.1. Detailed settings

For evaluation, LFW-SUB is divided into 10 predefined splits. Each time nine of them are used for CNN-M training and the remaining one is used for testing, according to [41]. The face images are detected by MTCNN [44] and mapped to a face template of size 58×58 based on 5 facial landmarks (two eyes, two mouse corners and a nose tip). The performance is evaluated by the estimated mean accuracy $\hat{\mu}$ and the standard error of the mean S_E:

$$\hat{\mu} = \frac{\sum_{i=1}^{10} p_i}{10}, \; S_E = \sqrt{\frac{\sum_{i=1}^{10}(p_i - \hat{\mu})^2}{90}}, \tag{9}$$

where p_i is the perception of correct classification, using i-th fold for testing, which is described in [29]. The cosine metric and threshold comparison are used

for computing p_i for both face verification and identification throughout the article.

4.2.2. Results

According to the detailed settings, the final results are listed in Table 4.

Table 4: Results on LFW-SUB database.

Model	$\hat{\mu} \pm S_E$ (%)
Baseline [39]	78.95 ± 0.36
CNN-M-$\mathcal{L}_S$	78.18 ± 0.46
CNN-M-$\mathcal{L}_C$	90.25 ± 0.29
CNN-M-$\mathcal{L}_{CD}$	90.97 ± 0.31
CNN-M-$\mathcal{L}_{ACD}$	$\mathbf{91.25 \pm 0.33}$

It can be seen that both CNN-M-$\mathcal{L}_{CD}$ and CNN-M-$\mathcal{L}_{ACD}$ perform better than CNN-M-$\mathcal{L}_S$ and CNN-M-$\mathcal{L}_C$. Especially, CNN-M-$\mathcal{L}_{ACD}$ increases 1% compared to CNN-M-$\mathcal{L}_C$, indicating that giving clear restraint for the inter-class variations is necessary to learn more compact and discriminative face features. Further, CNN-M-$\mathcal{L}_{ACD}$ performs better than CNN-M-$\mathcal{L}_{CD}$, implying that $\mathcal{L}_{acd}$ can give a better description of the relationship with different classes and make the center features better adapt to the feature learning than $\mathcal{L}_{cd}$. In a word, our proposed losses can learn more discriminative face features for CNN-M than the other two losses.

4.3. Large-scale face verification

For further evaluating the effectiveness of the proposed losses, we design experiments based on ResNet-50 and large-scale training CASIA-WebFace database, the face verification tasks include three different levels of Image-to-Image Face Verification (IIFV), a Video-to-Video Face Verification (VVFV) and a Template-to-Template Face Verification (TTFV), which will be detailed in the following.

4.3.1. Detailed settings

We use MTCNN [44] to detect the CASIA-WebFace database, and the obtained face images are aligned by five facial landmarks (locations of two eye centers, two mouth corners and a nose tip) with a given face template. After detection and alignment, we finally obtain about 0.49M face images for training. In addition, we keep the detection and alignment of the training database and the testing database the same for each task. The face features were extracted from ResNet-50[5] trained by compared supervisory signals, namely, $\mathcal{L}_S$, $\mathcal{L}_C$, $\mathcal{L}_{CD}$ and $\mathcal{L}_{ACD}$. And we denote the final obtained best CNN models by ResNet-$\mathcal{L}_S$, ResNet-$\mathcal{L}_C$, ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$, respectively.

For large-scale face verification on LFW, the standard LFW protocol is chosen for evaluation. Actually, the standard LFW protocol is very limited. According to Deng *et al.* [37], the impostor matches are very easy since the natural inter-class variance are large on LFW. Liao *et al.* [42] also claimed that the performance on LFW may be too optimistic because the underlying false accept rate may still be high, and performance evaluation at low FARs is not statistically sound by the standard protocol due to limited number of impostor matches. Based on these, we extend our IIFV tasks on FGLFW database, and also conduct IIFV on LFW according to the BLUFR [42] by considering more impostor matches. Further, we conduct experiments on YTF for the more challenging VVFV. For evaluation, we randomly select 100 pairs of frames per video and use the average cosine similarity of 100 pairs as the similarity of a test video pair. In the last, we choose a more challenging IJB-A database to evaluate the performance of the proposed approaches on TTFV.

4.3.2. LFW evaluation

The face verification is to determine whether one of the given $6,000$ face pairs is belong to the same identity or not, where the $6,000$ face pairs consist

[5]We fine tune the ResNet-50 with the compared supervision signals on a pretrained CNN model, which achieves 97.67% accuracy on LFW.

of $3,000$ genuine matches and $3,000$ impostor matches for classification. We report the performance of $\hat{\mu}$ and S_E in Table 5 for comparison. Namely, we list the state-of-the-art methods for IIFV in the first part of Table 5, and compare the performance of the most related methods under the same experiment environment in the second part of Table 5 for fairness.

Table 5: Comparing performance (%) on LFW.

Method	#Train	#Model	$\hat{\mu}$ ($\pm$ S_E)
FaceNet [25]	200M	1	99.63
SphereFace [32]	0.49M	1	99.42
Center Approach [28]	0.7M	1	99.28
NormFace [33]	1.5M	1	99.19
DeepID2 [23]	0.2M	200	99.15
VGG [18]	2.6M	1	98.95
L-Softmax [27]	0.49M	1	98.71
WebFaceCNN [39]	0.49M	1	97.73
DeepID [20]	0.2M	1	97.45
DeepFace [24]	4M	3	97.35
ResNet-$\mathcal{L}_S$	0.49M	1	98.67 $\pm$ 0.16
ResNet-$\mathcal{L}_C$	0.49M	1	98.92 $\pm$ 0.14
ResNet-$\mathcal{L}_{CD}$	0.49M	1	**99.10** $\pm$ **0.13**
ResNet-$\mathcal{L}_{ACD}$	0.49M	1	99.08 $\pm$ 0.14

From the first part of the table, we can see that the CNN models supervised by the proposed losses still have room for improvement compared to the state-of-the-art, such as Center Approach with more training data, NormFace with cosine similarity optimization, and SphereFace with both careful designed CNN architecture and cosine similarity optimization. What is acceptable is that, the CNN models supervised by the proposed losses performs better than several state-of-the-art methods on LFW, such as VGG, L-Softmax, WebFaceCNN, DeepID and DeepFace. Besides, the second part of the table shows that the CNN models supervised by the proposed losses perform much better than the related models for IIFV. Particularly, ResNet-$\mathcal{L}_{ACD}$ performs best and ResNet-$\mathcal{L}_{CD}$ gets the second place.

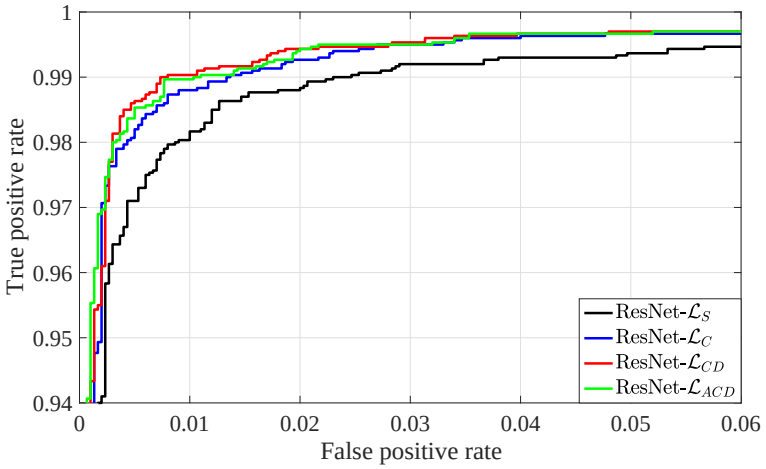

Fig. 4: ROC curves on LFW database for related models.

In addition, we also illustrate the corresponding ROC curves for related method in Fig. 4. Specially, the true positive rate for ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ almost surpass ResNet-$\mathcal{L}_C$ by clear margin in $(0, 0.02)$. These all show the superiority of proposed approach.

4.3.3. FGLFW evaluation

Since FGLFW only modifies the negative face pairs defined in the standard LFW protocol, the testing paradigms of LFW can be directly used. Similarly, we list the state-of-the-art results and also report our final performance in Table 6 and Fig. 5.

Compared with the first part of Table 6, ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ perform better than human performance, and also achieve better accuracy than the state-of-the-art methods, such as DCMN, VGG, DeepFace and DeepID2. Specifically, when it comes to comparison under the same experimental environment, the second part of the table shows that the CNN models supervised by the proposed losses perform best. Particularly, ResNet-$\mathcal{L}_{CD}$ performs best, surpassing the baseline ResNet-$\mathcal{L}_S$ by 2.46%; ResNet-$\mathcal{L}_{ACD}$ gets the second place, and also surpassing the third place ResNet-$\mathcal{L}_C$ by 1.05%.

Table 6: Comparing performance (%) on FGLFW.

Method	#Train	$\hat{\mu}$ ($\pm$ S_E)
Noisy Softmax [45]	0.5M	94.50
Human [37]	n/a	92.00
DCMN [37]	0.5M	91.00
VGG [18, 37]	2.6M	85.78
DeepFace [24, 37]	0.5M	78.78
DeepID2 [21, 37]	0.2M	78.25
ResNet-$\mathcal{L}_S$	0.49M	91.72$\pm$ 0.35
ResNet-$\mathcal{L}_C$	0.49M	93.02 $\pm$0.43
ResNet-$\mathcal{L}_{CD}$	0.49M	**94.18 $\pm$0.22**
ResNet-$\mathcal{L}_{ACD}$	0.49M	94.07 $\pm$ 0.27

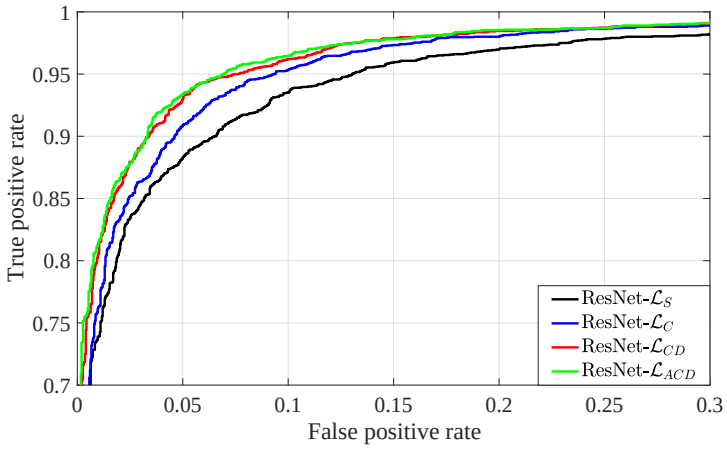

Fig. 5: ROC curves on FGLFW database for related models.

Besides, the ROC curves in Fig. 5 show that ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ surpass ResNet-$\mathcal{L}_C$ by clear margin in $(0, 0.15)$. These all show the superiority of proposed approach. These show that the effect of decreasing the intra-class variations and increasing the inter-class variations in CD loss and ACD loss is significant for further enhancing the discrimination of the face feature for more challenging IIFV.

4.3.4. BLUFR evaluation

BLUFR is a more challenging protocol that containing both verification and open-set identification scenarios, it is designed to fully exploit all the $13,233$ LFW face images for large-scale unconstrained face recognition evaluation, with a focus at low FARs. It introduces 10 trials of experiments, with each trial containing about $156,915$ genuine matching scores and $46,960,863$ impostor matching scores on average for performance evaluation.

According to [42], we report the mean verification rates (%) at the false accept rate of 0.1% (or 1%) subtracted by the corresponding standard deviations over 10 trials. The results are displayed in Table 7. From the table, we see clearly that ResNet-$\mathcal{L}_{ACD}$ achieves the best performance on both two evaluations. And ResNet-$\mathcal{L}_{CD}$ also performs better than both ResNet-$\mathcal{L}_S$, and ResNet-$\mathcal{L}_C$. ResNet-$\mathcal{L}_{ACD}$ surpasses ResNet-$\mathcal{L}_C$ on the two cases by 2.07% and 0.35%, respectively.

Table 7: Performance (%) for the verification scenario of BLUFR evaluation.

Method	FAR=0.1%	FAR=1%
NormFace [33]	95.83	-
Center Approach[28, 33]	93.35	-
LightenedCNN [46]	89.12	-
WebFaceCNN BaseLine [39]	80.26	-
HD-LBP + JB [42]	41.66	65.84
HD-LBP + LDA [42]	36.12	61.39
ResNet-$\mathcal{L}_S$	87.14	96.58
ResNet-$\mathcal{L}_C$	90.10	97.86
ResNet-$\mathcal{L}_{CD}$	91.50	97.95
ResNet-$\mathcal{L}_{ACD}$	**92.17**	**98.21**

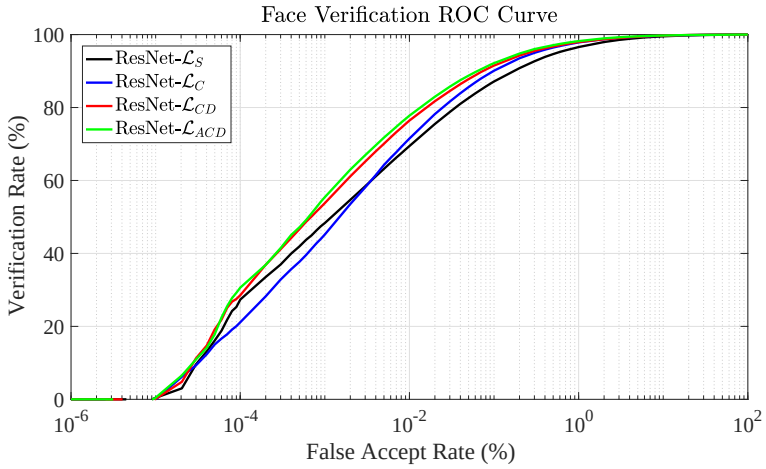

Fig. 6: Verification ROC curves for related models under the BLUFR protocol.

Besides, we illustrate the face verification ROC curves in Fig. 6 for better comparison. The figure shows that ResNet-$\mathcal{L}_{ACD}$ and ResNet-$\mathcal{L}_{CD}$ get the better verification rate than the other two compared methods almost from $10^{-4}\%$ to $10^{2}\%$. These all demonstrate the necessary of restraining both the intra-class variations and the inter-class variations. In a word, the experiments show that the proposed approaches are more suitable to learn compact and discriminative face features for IIFV.

4.3.5. YTF evaluation

For VVFV on YTF, we obey the protocol described in [29, 39], and report the estimated mean accuracy $\hat{\mu}$ and the standard error of the mean S_E for comparison. The comparison with the most recent state-of-the-art on the two datasets is given in Table 8.

As shown in Table 8, ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ achieve the accuracy of 93.54% and 93.50%, respectively. They perform not only better than ResNet-$\mathcal{L}_S$ and ResNet-$\mathcal{L}_C$ in the same experiment environment, but also perform better than the state-of-the-art WebFaceCNN and DeepFace, with more less training data. The ROC curves in Fig. 7 also show that ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ surpass ResNet-$\mathcal{L}_C$ almost in $(0, 0.2)$. The phenomena illuminate that

the reliability of the proposed $\mathcal{L}_{CD}$ and $\mathcal{L}_{ACD}$ and show that they are effective for the more challenging TTFV.

Table 8: Comparing performance (%) on YTF.

Method	#Train	#Model	$\hat{\mu}$ ($\pm$ S_E)
VGG [18]	2.6M	1	97.3
DCFL [47]	4.7M	1	96.06
FaceNet [25]	200M	1	95.1
Center Approach [28]	0.7M	1	94.9
NormFace [33]	1.5M	1	94.72
DeepID2+ [23]	0.3M	25	93.2
WebFaceCNN [39]	0.49M	1	92.24
DeepFace [24]	4M	3	91.4
ResNet-$\mathcal{L}_S$	0.49M	1	92.88$\pm$ 0.42
ResNet-$\mathcal{L}_C$	0.49M	1	93.38 $\pm$0.36
ResNet-$\mathcal{L}_{CD}$	0.49M	1	**93.54 $\pm$0.41**
ResNet-$\mathcal{L}_{ACD}$	0.49M	1	93.50 $\pm$ 0.33

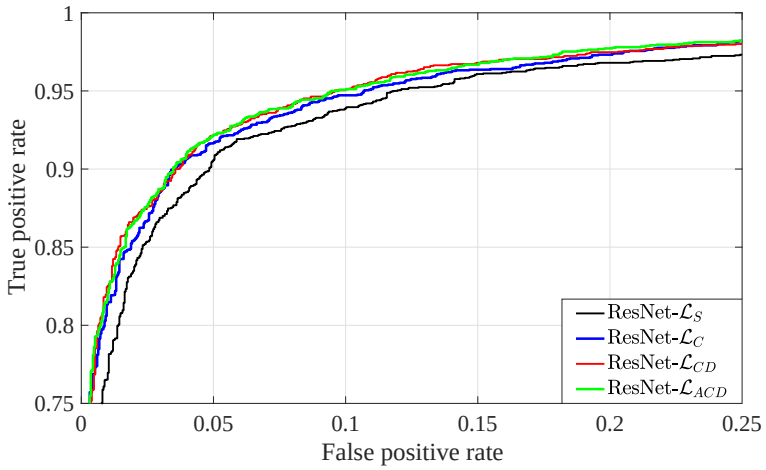

Fig. 7: ROC curves on YTF database for related models.

4.3.6. IJB-A evaluation

For TTFV on IJB-A, we obey the protocol described in [43], and report the value of True Accept Rate (TAR) at a fixed False Accept Rate (FAR) for comparison. The comparisons with the state-of-the-art and related models are given in Table 9.

Table 9: Verification performance (%) on IJB-A for related models.

Method	TAR@FAR=0.1	TAR@FAR=0.01	TAR@FAR=0.001
DA-GAN [48]	99.1	97.6	93
Template Adaptation [49]	97.9	93.9	83.6
All-In-One Face [50]	97.6	92.2	82.3
VGG-Face [26, 50]	-	80.5	-
DCNN [51]	96.7	83.8	-
GOTS [43]	62.7	40.6	19.8
ResNet-$\mathcal{L}_S$	60.11	32.26	20.62
ResNet-$\mathcal{L}_C$	**63.93**	36.56	**21.04**
ResNet-$\mathcal{L}_{CD}$	63.80	**39.38**	20.67
ResNet-$\mathcal{L}_{ACD}$	61.21	37.57	20.89

From the table, the performance of the proposed approaches still can't reach the high level of the state-of-the-art, the reason will be described in Subsection 4.5. However, the proposed approaches can achieve respective performance under the same experiment setting in the second part of the table. The proposed approaches need to be further improved for more challenging TTFV tasks.

The five different experiments on face verification, including three different levels of IIFV tasks, a VVFV task and a TTFV task, demonstrate that decreasing the intra-class variations and increasing the inter-class variations simultaneously is significant to learn compact and discriminative face features for more difficult face verification tasks and get relatively more stable performance.

4.4. Face identification

In this section, we focus on the more challenging face identification task, which aims to search for a person's face in a set of enrolled images or templates. For evaluation, we select LFW (BLUFR protocol) and IJB-A database as the testing benchmarks, which are really more challenging than the mentioned four face verification tasks.

4.4.1. Detailed setting

For BLUFR evaluation, it is based on 10 random trials of face identification tasks, which fully exploit all the $13,233$ face images in LFW. Specially, in each test trial, there are $1,000$ subjects to constitute the gallery set, about $4,350$ face images of $1,000$ subjects to constitute the genuine probe set, and about $4,357$ images of $3,249$ subjects to constitute the impostor probe set. According to [42], the mean detection and identification rates (%), close-set face identification CMC curves and open-set face identification CMC curves at FAR= 1% are used for reporting the performance of the related models.

4.4.2. BLUFR evaluation

We report the the mean detection and identification rates (%) at Rank 1 subtracted by the corresponding standard deviations over 10 trials in Table 10.

Table 10: Performance (%) for the identification scenario of the BLUFR protocol.

Method	FAR=1%	FAR=10%
NormFace [33]	77.18	-
Center Approach[28, 33]	67.86	-
LightenedCNN [46]	61.79	-
WebFaceCNN [39]	28.9	-
HD-LBP + JB [42]	18.07	32.63
HD-LBP + LDA [42]	14.94	31.39
ResNet-$\mathcal{L}_S$	52.77	75.51
ResNet-$\mathcal{L}_C$	54.63	78.84
ResNet-$\mathcal{L}_{CD}$	60.50	**81.9**
ResNet-$\mathcal{L}_{ACD}$	**60.54**	81.23

It shows that ResNet-$\mathcal{L}_{ACD}$ and ResNet-$\mathcal{L}_{CD}$ perform better than the other two models by clear margins on both the two levels of evaluations. They even surpass ResNet-$\mathcal{L}_C$ by about 6% when FAR = 1% and surpass ResNet-$\mathcal{L}_C$ by more than 2% when FAR = 10%. These show the superiority of the proposed approach on face identification tasks.

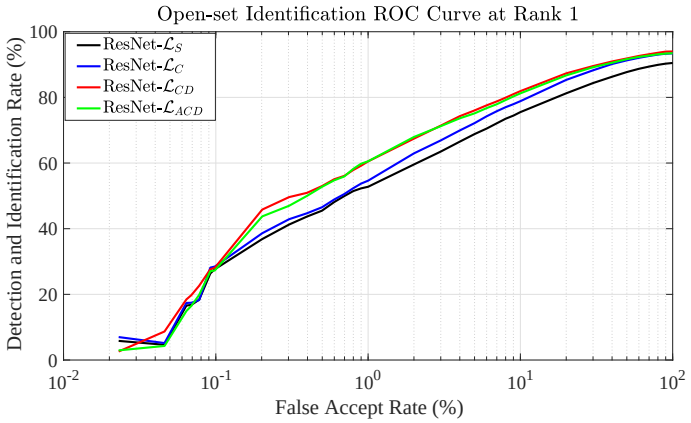

Fig. 8: Closed-set face identification ROC curves for related models.

In addition, we illustrate the ROC curves at Rank 1 in Fig. 8, and show the CMC curves to measure the closed-set identification performance in Fig. 9 for

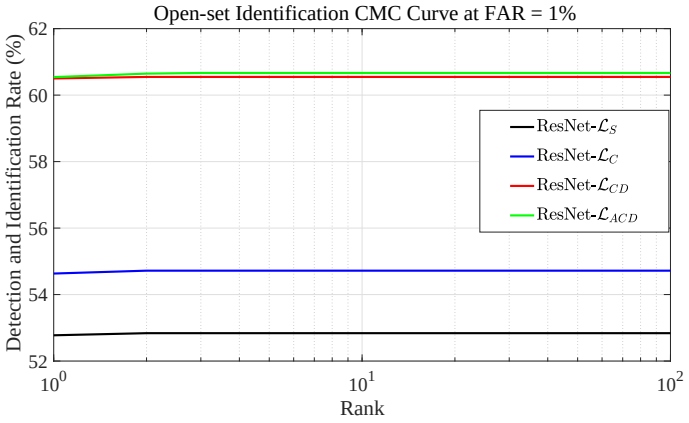

Fig. 9: Open-set identification CMC curves at FAR=1% for related models.

a better comparison. The ROC performance in Fig. 8 shows that the curves corresponding to ResNet-$\mathcal{L}_{ACD}$ and ResNet-$\mathcal{L}_{CD}$ surpass the others by clear margins almost from $10^{-1}\%$ to $10^{2}\%$. CMC curves in Fig. 9 also shows the superior performance of ResNet-$\mathcal{L}_{ACD}$ and ResNet-$\mathcal{L}_{CD}$. These all demonstrate the proposed approach helps to learn more compact and discriminative face features for face recognition.

4.4.3. IJB-A evaluation

We report the overall face identification performance for related CNN models in Table 11.

Table 11: Face identification performance (%) on the IJB-A dataset.

Method	CMC (%)			TPIR@FPIR's of (%)	
	Rank-1	Rank-5	Rank-10	0.1	0.01
DA-GAN [48]	97.1	98.9	-	94.9	89
NAN [52]	95.8	-	98.6	91.7	81.7
All-In-One Face [50]	94.7	-	98.8	88.7	79.2
CNN_{media}+TPE [53]	93.2	-	97.7	86.3	75.3
Template Adaptation [49]	92.8	-	98.6	88.2	77.4
VGG-Face [26, 50]	91.3	-	98.1	67	46
Multi-Pose Face [54]	85.8	93.8	-	-	-
DCNN [55]	85.2	-	95.4	-	-
GOTS [43]	44.3	59.5	-	23.5	4.7
ResNet-$\mathcal{L}_S$	84.41	95.70	97.64	63.80	**39.38**
ResNet-$\mathcal{L}_C$	85.97	96.39	98.03	60.11	32.26
ResNet-$\mathcal{L}_{CD}$	87.03	96.64	**98.24**	**64.47**	36.62
ResNet-$\mathcal{L}_{ACD}$	**87.27**	**96.91**	98.21	63.93	36.56

The table shows that ResNet-$\mathcal{L}_{CD}$ and ResNet-$\mathcal{L}_{ACD}$ still can't reach the high level of the careful designed template-to-template methods shown in the first part of the table. However, they still give the respectable performance when compared to several methods in the first part of the table on Rank 1, such as Multi-Pose Face, DCNN and GOTS. For the same experimental environment in the second part of the table, ResNet-$\mathcal{L}_{ACD}$ achieves the best performance at Rank 1 and Rank 5, and gets the second place at Rank 10. ResNet-$\mathcal{L}_{CD}$ surpasses ResNet-$\mathcal{L}_C$ by 1.06%, 0.25% and 0.21% in Rank 1, Rank 5 and Rank 10, respectively. These all show the effectiveness of the proposed approaches.

In short, experiments on the two face identification tasks show that the proposed approaches are also effective to learn more compact and discriminative face representations for identification.

4.5. Discussions

The proposed approaches are verified to be effective in most experiments of the preceding subsections. However, they still can't reach the level of the

state-of-the-art when evaluated on IJB-A database in both face verification and face identification. We take All-in-one Face method [50] for example to give the limitations of the proposed approaches:

1. Inaccurate face detection: we use MTCNN (v1) for detection [44] and use 5 facial points for alignment, which results in $1,088$ undetected faces. In contrast, All-In-One Face [50] uses 6 fiducial point extraction by Hyper-Face method [56].

2. None usage of the training splits in IJB-A dataset for CNN training: we only use the training data to compute the threshold for the testing split. In contrast, All-in-one Face takes advantage of the training splits in IJB-A for CNN training.

3. Unsuitable definition of final representation of a given template: We just use the average of the features of a given template for the final representation. In contrast, All-in-one Face flattens the template features by media pooling.

It should be noticed that we use no template adaptation method for reporting the final performance. The purpose of presentation of IJB-A face recognition is only to verify the effectiveness of the proposed approaches. We will pay more attention to the phenomenon of non-optimistic performance of the proposed approaches on IJB-A in the further.

5. Conclusions

In this paper, we have presented two novel loss functions, referred to as Compact Discriminative loss and Advanced Compact Discriminative loss. They reduce the intra-class variations and enlarge the inter-class variations simultaneously by forcing the feature to be close to the real class and escape from the misclassified class. We evaluate the performance of the two losses on several typical CNN architectures, and compare them with the state-of-the-art losses

on several famous face verification tasks and face identification tasks. Various evaluation implementations show that the proposed losses help to result in more compact and discriminative features for face recognition.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Number: 61801325) and the Natural Science Foundation of Tianjin City (Grant Number: 18JCQNJC00600), and the Fundamental Research Funds for the Central Universities. The authors would like to thank the referees for their constructive suggestions.

References

[1] T. Ahonen, A. Hadid, M. Pietikainen, Face description with local binary patterns: Application to face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (12) (2006) 2037–2041.

[2] L. Shen, L. Bai, A review on gabor wavelets for face recognition, Pattern Analysis and Applications 9 (2-3) (2006) 273–292.

[3] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[4] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13 (1) (1967) 21–27.

[5] K. Fukunaga, P. M. Narendra, A branch and bound algorithm for computing k-nearest neighbors, IEEE Transactions on Computers 100 (7) (1975) 750–753.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 210–227.

[7] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: Which helps face recognition?, in: IEEE International Conference on Computer Vision Workshops, IEEE, 2011, pp. 471–478.

[8] P. Zhu, W. Zuo, L. Zhang, S. C.-K. Shiu, D. Zhang, Image set-based collaborative representation for face recognition, IEEE Transactions on Information Forensics and Security 9 (7) (2014) 1120–1132.

[9] Z.-M. Li, Z.-H. Huang, K. Shang, A customized sparse representation model with mixed norm for undersampled face recognition, IEEE Transactions on Information Forensics and Security 11 (10) (2016) 2203 – 2214.

[10] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.

[11] C. Ding, D. Tao, Robust face recognition via multimodal deep face representation, IEEE Transactions on Multimedia 17 (11) (2015) 2049–2058.

[12] C. C. Pham, J. W. Jeon, Robust object proposals re-ranking for object detection in autonomous driving using convolutional neural networks, Signal Processing: Image Communication 53 (2017) 110–122.

[13] Z.-L. Chen, J. Wang, W.-J. Li, N. Li, H.-M. Wu, D.-W. Wang, Convolutional neural network with nonlinear competitive units, Signal Processing: Image Communication 60 (2018) 193–198.

[14] B. Wu, Z. Chen, J. Wang, H. Wu, Exponential discriminative metric embedding in deep learning, Neurocomputing.

[15] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[16] M. Lin, Q. Chen, S. Yan, Network in network, ArXiv Preprint ArXiv:1312.4400.

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 1–9.

[18] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, ArXiv Preprint ArXiv:1409.1556.

[19] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2016, pp. 770–778.

[20] Y. Sun, X. Wang, X. Tang, Deep learning face representation from predicting 10,000 classes, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014, pp. 1891–1898.

[21] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Advances in Neural Information Processing Systems, 2014, pp. 1988–1996.

[22] Y. Sun, D. Liang, X. Wang, X. Tang, Deepid3: Face recognition with very deep neural networks, ArXiv Preprint ArXiv:1502.00873.

[23] Y. Sun, X. Wang, X. Tang, Deeply learned face representations are sparse, selective, and robust, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 2892–2900.

[24] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014, pp. 1701–1708.

[25] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 815–823.

[26] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al., Deep face recognition., in: BMVC, Vol. 1, 2015, p. 6.

[27] W. Liu, Y. Wen, Z. Yu, M. Yang, Large-margin softmax loss for convolutional neural networks., in: International Conference on Machine Learning, 2016, pp. 507–516.

[28] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 499–515.

[29] G. B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Tech. rep., Tech. Rep 07-49, Dept. Comput. Sci., University of Massachusetts, Amherst, MA, USA (2007).

[30] Y. Zhang, K. Shang, J. Wang, N. Li, M. M. Y. Zhang, Patch strategy for deep face recognition, IET Image Processing 12 (5) (2018) 819–825.

[31] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Web-scale training for face identification, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 2746–2754.

[32] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, L. Song, Sphereface: Deep hypersphere embedding for face recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2017, pp. 212–220.

[33] F. Wang, X. Xiang, J. Cheng, A. L. Yuille, Normface: L2 hypersphere embedding for face verification, in: ACM International Conference on Multimedia, ACM, 2017, pp. 1041–1049.

[34] M. M. Zhang, Y. Xu, H. Wu, Orientation truncated centre learning for deep face recognition, Electronics Letters 54 (19) (2018) 1110–1112.

[35] M. M. Zhang, K. Shang, H. Wu, Learning deep discriminative face features by customized weighted constraint, Neurocomputing 332 (2019) 71–79.

[36] G. B. Huang, E. Learned-Miller, Labeled faces in the wild: Updates and new reporting procedures, Tech. rep., Tech. Rep 14-003, Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA (2014).

[37] W. Deng, J. Hu, N. Zhang, B. Chen, J. Guo, Fine-grained face verification: Fglfw database, baselines, and human-dcmn partnership, Pattern Recognition 66 (2017) 63–73.

[38] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 529–534.

[39] D. Yi, Z. Lei, S. Liao, S. Z. Li, Learning face representation from scratch, ArXiv Preprint ArXiv:1411.7923.

[40] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: ACM International Conference on Multimedia, ACM, 2014, pp. 675–678.

[41] G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, T. Hospedales, When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition, in: IEEE International Conference on Computer Vision Workshops, 2015, pp. 142–150.

[42] S. Liao, Z. Lei, D. Yi, S. Z. Li, A benchmark study of large-scale unconstrained face recognition, in: IEEE International Joint Conference on Biometrics, IEEE, 2014, pp. 1–8.

[43] 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: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1931–1939.

[44] K. Zhang, Z. Zhang, Z. Li, Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, IEEE Signal Processing Letters 23 (10) (2016) 1499–1503.

[45] B. Chen, W. Deng, J. Du, Noisy softmax: Improving the generalization ability of dcnn via postponing the early softmax saturation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[46] X. Wu, R. He, Z. Sun, A lightened cnn for deep face representation, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 4, 2015.

[47] W. Deng, B. Chen, Y. Fang, J. Hu, Deep correlation feature learning for face verification in the wild, IEEE Signal Processing Letters 24 (12) (2017) 1877–1881.

[48] J. Zhao, L. Xiong, P. K. Jayashree, J. Li, F. Zhao, Z. Wang, P. S. Pranata, P. S. Shen, S. Yan, J. Feng, Dual-agent gans for photorealistic and identity preserving profile face synthesis, in: Advances in Neural Information Processing Systems, 2017, pp. 66–76.

[49] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, A. Zisserman, Template adaptation for face verification and identification, in: IEEE Conference on Automatic Face and Gesture Recognition, IEEE, 2017, pp. 1–8.

[50] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, R. Chellappa, An all-in-one convolutional neural network for face analysis, in: IEEE Conference on Automatic Face and Gesture Recognition, IEEE, 2017, pp. 17–24.

[51] J.-C. Chen, R. Ranjan, A. Kumar, C.-H. Chen, V. M. Patel, R. Chellappa, An end-to-end system for unconstrained face verification with deep convolutional neural networks, in: IEEE International Conference on Computer Vision Workshops, 2015, pp. 118–126.

[52] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, G. Hua, Neural aggregation network for video face recognition., in: CVPR, Vol. 4, 2017, p. 7.

[53] S. Sankaranarayanan, A. Alavi, C. D. Castillo, R. Chellappa, Triplet probabilistic embedding for face verification and clustering, in: IEEE International Conference on Biometrics: Theory, Applications, and Systems, 2016, pp. 1–8.

[54] X. Yin, X. Liu, Multi-task convolutional neural network for pose-invariant face recognition, IEEE Transactions on Image Processing.

[55] J.-C. Chen, V. M. Patel, R. Chellappa, Unconstrained face verification using deep cnn features, in: IEEE International Conference on Computer Vision, IEEE, 2016, pp. 1–9.

[56] R. Ranjan, V. M. Patel, R. Chellappa, Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence PP (99) (2016) 1–1.