Learning Deep Discriminative Face Features by Customized Weighted Constraint

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

[a] *Center for Combinatorics, LPMC, Nankai University, Tianjin, 300071, PR China*
[b] *College of Mathematics and Econometrics, Hunan University, Changsha, Hunan 410082, PR China.*
[c] *Center for Applied Mathematics, Tianjin University, Tianjin,300072, PR China*

Abstract

Different convolutional neural networks (CNNs) may learn different levels of discriminative features to represent the raw face data. To enhance the discrimination of deeply learned face features, we propose a customized weighted discriminative loss (CWD) to seek a customized constraint for mitigating the large perturbations caused by imbalanced distribution of correctly-classified features and mis-classified features. It focuses on mapping the raw data into a feature space such that deeply learned face features can achieve a high discrimination for representation, by retraining the intra-class variations and the inter-class variations, simultaneously. Extensive experiments carried out on several famous face recognition benchmarks, including LFW, YTF, FGLFW and BLUFR, demonstrate that the proposed approach can achieve superior performance over the related approaches.

Keywords: Face recognition, Customized weighted discriminative loss, Customized weighted constraint, Convolutional neural network

1. Introduction

Face recognition has been one of the most challenging and attractive studied topics of computer vision. Accurate face recognition depends on high-quality

*Corresponding author.
Email address: kunzzz.shang@gmail.com (Kun Shang)

face representation, which should be discriminative for inter-personal variations, and be discriminative for the intra-personal variations, simultaneously. However, conventional face representations are built on local descriptors, which are too shallow to differentiate the complicated nonlinear facial appearance variations, such as pose, illumination, expression and occlusion. The complicated facial appearance variations call for more advanced techniques for robustness face representation. Recently, deep learning [1] has achieved impressive results in computer vision applications, including action recognition [2], object segmentation [3], object tracking [4, 5], attention prediction [6, 7], photo cropping[8], semantic segmentation [9], motion segmentation [10] and salient object detection [11, 12, 13, 14]. Further, the face features based on deep learning has achieved phenomenal performance for robustness face representation [15, 16, 17, 18].

Taigman *et al.* [19] proposed the DeepFace system, which used Softmax loss as the supervisory signal to train CNN model and achieved 97.35% on LFW database [20], approaching to the human-level 97.53%. The authors later extended this work in [21], by increasing the size of the training database to 10 million subjects with 50 images each on average. They proposed a bootstrapping strategy to select training identities that consist of both easy and hard samples to avoid the saturation existed in CNN. Meanwhile, Sun *et al.* [22, 23, 24] proposed the DeepID series of papers, each of which steadily increased the performance of face recognition on LFW database. Particularly, a number of new ideas were incorporated over the series of papers. DeepID [22] used multiple CNNs to get fusion face features and applied Bayesian learning framework [25] to get the suitable metric. DeepID2 [23] combined the Face Identification loss and Face Verification loss for more effective training and empirically verified that the combined supervisory signal is helpful to promote the power of CNN to extract discriminative features. DeepID2+ [24] considered to increase the dimension of hidden representations to achieve new state-of-the-art performance on both LFW and YTF benchmarks [26].

Inspired by designing effective supervisory signal for CNN in DeepID2+ [24], Schroff *et al.* [27] introduced Triplet loss for FaceNet system, which used

nearly 100M-200M training faces consisting of about 8M different identities for training the powerful CNN models. Further, Parkhi *et al.* [28] focused on how to collect very large scale face databases (such as 2.6 million) and how to construct effective CNN architectures. As a supplement, Hu *et al.* [29] investigated the influence of different CNN architectures and tested different implementation choices for extracting face features. All these researchers contribute to give a better understanding and innovate ideas to promote the development of face recognition.

Recently, designing suitable loss functions for extracting CNN face features has achieved great success. For example, Wen *et al.* [30] used the joint supervision of Softmax loss and Center loss for training CNN face features. Wang *et al.* [31] inherited the idea of Center loss and studied the effect of normalization during training and optimized cosine similarity instead of inner-product. Liu *et al.* [32] proposed L-Softmax to employ a margin constraint for Softmax loss to achieve a classification angle margin between classes. Liu *et al.* [33] later extended L-Softmax loss by considering the cosine normalization. These works all achieved excellent performance on face recognition by adopting well-designed CNN architectures.

Among these effective loss based approaches, CenterApproach [30] that focusing on minimizing the intra-class variations between each feature and its corresponding class center, has achieved great success for addressing face recognition problems with only 0.7M training data. However, mentioned by [30], there exist large perturbations caused by few mislabeled samples. Specially, when the number of the mis-classified features is much more than the number of the correctly-classified features, it causes poor separability. On the contrary, if most of the correctly-classified features are not so close to the center feature, even the number of correctly-classified features surpasses the number of the mis-classified features, poor compactness will emerge. Further, if these perturbations are not given enough treatment, it may result in less discrimination for face representation. How to better mitigate the large perturbation (such as poor separable distribution and poor compact distribution) and better enhance

the performance of CNN for extracting more discriminative face features, is still a challenging problem.

In this paper, we propose the customized weighted discriminative (CWD) loss to address the large perturbation problem. The major contributions of this paper are summarized as follows:

- We analyze the large perturbations from the aspect of mis-classified features and correctly-classified features during the training, and propose a CWD loss to supervise the learning of CNN for getting more discriminative face features, by seeking a customized weighted constraint for the two kinds of features.

- We use a toy example for showing the phenomenon of alleviating the perturbations of the features. Detailed analysis are also reported on issues such as the limitations and future directions of the proposed approach.

- We evaluate the performance of the proposed approach on LFW, YTF, FGLFW and BLUFR benchmarks, the experimental results show that the proposed approach can achieve promising performance for face recognition.

2. The proposed approach

2.1. Center loss and motivation

As mentioned before, CenterApproach [30] is a simple and trainable approach for addressing face recognition problems. It takes advantage of the Center loss $\mathcal{L}_C$ to characterize the intra-class variations by summing the distance between each feature and its corresponding class center, where

$$\mathcal{L}_C = \frac{1}{2M}\sum_{i=1}^{M}\|\boldsymbol{x}_i - \boldsymbol{c}_{l_i}\|^2, \tag{1}$$

M is the mini-batch size, l_i is the corresponding label for feature $\boldsymbol{x}_i$, $\boldsymbol{c}_{l_i}$ is the class center for l_i-th class. By minimizing $\mathcal{L}_C$, the intra-class variations of the

deeply learned face features can be decreased during the training, and the CNN models can be restrained to obtain more discriminative face features.

However, according to [30], there exist large perturbations caused by few mislabeled samples. If these mislabeled samples are not given enough treatment, poor separability or poor separability may occur when there are much more mis-classified features or much more correctly-classified features, respectively. In such cases, the learned face features may not be so discrimination for face recognition. To address the issue, we propose a new method in the following to give more treatment for mentioned perturbations.

2.2. Customized weighted discriminative loss

The customized weighted discriminative (CWD) loss is to supervise CNNs for getting more discriminative learned features, which is formalized as

$$\mathcal{L}_{CWD} = \frac{1}{2M}\sum_{i=1}^{M}\tilde{d}(\boldsymbol{x}_i, \boldsymbol{c}_{l_i}, \boldsymbol{c}_{p_i}), \tag{2}$$

where M is the mini-batch size, $\tilde{d}(\boldsymbol{x}_i, \boldsymbol{c}_{l_i}, \boldsymbol{c}_{p_i})$ is a triplet distance defined as

$$\tilde{d}(\boldsymbol{x}_i, \boldsymbol{c}_{l_i}, \boldsymbol{c}_{p_i}) = \begin{cases} \tau \cdot d(\boldsymbol{x}_i, \boldsymbol{c}_{l_i}) & l_i = p_i, \\ (1-\tau) \cdot d(\boldsymbol{x}_i, \boldsymbol{c}_{l_i}) & l_i \neq p_i, \end{cases} \tag{3}$$

$\boldsymbol{x}_i$ is the feature for i-th sample, l_i is the label of $\boldsymbol{x}_i$, $\boldsymbol{c}_n$ is the class center for n-th class, p_i is the predicted label of $\boldsymbol{x}_i$, which is obtained from the softmax prediction in the last layer of a given CNN. And $d(\cdot, \cdot)$ is the pre-defined distance, we simply adopt the commonly used L2-distance, $\tau \in (0, 1)$ is a trade-off hyper-parameter. The triplet distance is designed for measuring the large perturbations caused by mis-classified features and correctly-classified features.

Actually, different proportions of the correctly-classified features ($l_i = p_i$) and the mis-classified features ($l_i \neq p_i$) may cause different levels of perturbations, which will influence on both the intra-class variations and the inter-class variations during the training. CWD loss introduces the hyper-parameter τ to constrain the perturbations for the feature distribution caused by the two kinds

of features. With a customized τ, CWD loss is expected to give suitable treatment for the perturbations. Namely, it aims to make the learned face features get away from the situation of poor separability or poor compactness, to better retrain the intra-class variations and the inter-class variations for learning highly discriminative face features.

We use the joint supervision of Softmax loss $\mathcal{L}_S$ and CWD loss $\mathcal{L}_{CWD}$ to train CNNs, by solving the following optimization objective

$$\boldsymbol{\theta}^* = \min_{\boldsymbol{\theta}} \mathcal{L}_S(X, Y, \boldsymbol{\theta}) + \lambda \mathcal{L}_{CWD}(X, Y, \boldsymbol{\theta}), \tag{4}$$

where

$$\mathcal{L}_S(X, Y, \boldsymbol{\theta}) = -\frac{1}{M} \sum_{i=1}^{M} \log \frac{e^{\boldsymbol{w}_{l_i}^\top \boldsymbol{x}_i + \boldsymbol{b}_{l_i}}}{\sum_{j=1}^{N} e^{\boldsymbol{w}_j^\top \boldsymbol{x}_i + \boldsymbol{b}_j}}, \tag{5}$$

X is the training data set, Y is the label data set, $\boldsymbol{\theta}$ is the parameter set, $\boldsymbol{w}_{l_i}$, $\boldsymbol{b}_{l_i}$ are parameters in last fully connected layer, λ is a trade-off hyper-parameter, N is the class number. The corresponding learning framework is shown in Fig. 1.

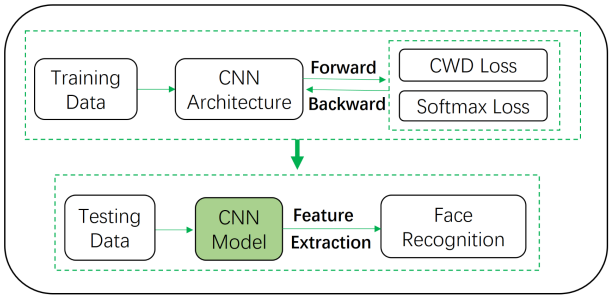


Fig. 1: Framework of face recognition based on the proposed approach.

The optimization objective (4) can be easily optimized by the standard stochastic gradient descent, according to (6) and (7), and the learning process is summarized in **Algorithm 1**.

$$\frac{\partial \mathcal{L}_{CWD}}{\partial \hat{x}_i} = \begin{cases} \frac{\tau}{M}(\hat{x}_i - c_{p_i}) & l_i = p_i, \\ \frac{1-\tau}{M} \cdot (\hat{x}_i - c_{p_i}) & l_i \neq p_i, \end{cases} \tag{6}$$

$$\frac{\partial \mathcal{L}_{CWD}}{\partial c_n} = \frac{\tau}{M} \sum_{l_m=n} (c_n - \hat{x}_{l_m}) + \frac{1-\tau}{M} \sum_{l_m \neq n} (c_n - \hat{x}_{l_m}). \tag{7}$$

Algorithm 1 Deep discriminative face features learning by customized constraint

Input: Training data set X, training label set Y. Initialized parameters $\boldsymbol{\theta}^t$, the n-th class center $\boldsymbol{c}_n^t$ and learning rate μ^t; hyper-parameter λ, τ and center learning rate γ, $t \leftarrow 0$.

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

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

1. $t = t + 1$;

2. Compute joint loss $\mathcal{L}_S(X, Y, \boldsymbol{\theta}^t) + \lambda \mathcal{L}_{CWD}(X, Y, \boldsymbol{\theta}^t)$;

3. Update $\boldsymbol{\theta}$ by $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \mu^t \sum_i [\frac{\partial \mathcal{L}_S^t}{\partial \boldsymbol{x}_i^t} + \lambda \frac{\partial \mathcal{L}_{CWD}^t}{\partial \boldsymbol{x}_i^t}] \frac{\partial \boldsymbol{x}_i^t}{\partial \boldsymbol{\theta}^t}$ according to (6);

4. Update $\boldsymbol{c}_n$ by $\boldsymbol{c}_n^{t+1} = \boldsymbol{c}_n^t - \lambda\gamma \frac{\partial \mathcal{L}_{CWD}^t}{\partial \boldsymbol{c}_n^t}$ according to (7);

end while

3. Experiments and results

In this section, we evaluate the effectiveness of the proposed approach for face recognition. For fair comparison, the softmax loss approach and the center loss approach are used as the baselines throughout the article. All experiments are implemented in the Caffe library [34] on Linux OS with the NVIDIA Tesla K80.

3.1. Implementation details

3.1.1. Databases

- **Data for visualization:** We use the database MNIST [35], a classical handwritten digit database with $60,000$ training examples and $10,000$ testing examples, for MNIST visualization.

- **Training data:** We use CASIA-WebFace database [36], it is a typical public face training database that containing $10,575$ subjects and $494,414$

images collected from the Internet. We choose it as the training database for obtaining CNN model because that it is almost independent of the LFW and YTF benchmark databases, and thus can dispel the chaos of evaluations.

- **Testing data:** We use several famous and challenging face recognition benchmarks to evaluate the effectiveness of the proposed approach for face feature extraction. The testing benchmarks are the Labeled Faces in the Wild (LFW) database [20], the Fine-grained LFW (FGLFW) database [37], YouTube Faces (YTF) database [26] and the Benchmark of Large-scale Unconstrained Face Recognition (BLUFR) [38]. The details will be described in the corresponding Subsection 3.3.1, Subsection 3.3.2, and Subsection 3.3.3, respectively.

For data preprocessing of the face recognition tasks, we keep the detection and alignment of the training database and the testing database the same for each task as [39, 40, 41] by using SeetaFace[1].

3.1.2. CNN architectures

We use two CNN architectures[2]: LeNet++ and ResNet-27 released by [30], shown in Table 1 and Table 2. LeNet++ is for MNIST visualization and to illustrate the effectiveness of the proposed approach for alleviating the perturbations of the learned features, which will be described in Subsection 3.2. The features are taken from fc4. ResNet-27 is for face feature extraction and to verify the effectiveness of the proposed approach on related face recognition tasks, which will be described in Subsection 3.3. The deep face features are taken from fc5 and the testing settings are same as [30].

Note: The $B(3,3)$ in Table 2 denotes a residual block composed of two 3×3 convolutional layers. For example, $B(3,3) \times 3$ and $B(3,3) \times 5$ denote 2 blocks in groups of convolutions and 5 blocks in groups of convolutions, respectively.

[1] [Online]. Available: `https://github.com/seetaface/SeetaFaceEngine`

[2] [Online]. Available: `https://github.com/ydwen/caffe-face`

Table 1: Architecture of LeNet++.

Name	Filter	Output Size
input	-	$28 \times 28 \times 3$
conv1a	5×5 conv, stride 1, pad 2	$28 \times 28 \times 32$
conv1b	5×5 conv, stride 1, pad 2	$28 \times 28 \times 32$
pool1	2×2 max-pool, stride 2	$14 \times 14 \times 32$
conv2a	5×5 conv, stride 1, pad 2	$14 \times 14 \times 64$
conv2b	5×5 conv, stride 1, pad 2	$14 \times 14 \times 64$
pool2	2×2 max-pool, stride 2	$7 \times 7 \times 64$
conv3a	5×5 conv, stride 1, pad 2	$7 \times 7 \times 128$
conv3b	5×5 conv, stride 1, pad 2	$7 \times 7 \times 128$
pool3	3×3 max-pool, stride 2	$3 \times 3 \times 128$
fc4	-	2
fc5	-	10

Table 2: Architecture of ResNet-27.

Name	Filter	Output Size
input	-	$112 \times 96 \times 3$
conv1a	3×3 conv, stride 1, pad 0	$110 \times 94 \times 32$
conv1b	3×3 conv, stride 1, pad 0	$108 \times 92 \times 64$
pool1	2×2 max-pool, stride 2	$54 \times 46 \times 3$
res1	$B(3,3) \times 2$	$54 \times 46 \times 3$
conv2	3×3 conv, stride 1, pad 0	$52 \times 44 \times 128$
pool2	2×2 max-pool, stride 2	$26 \times 22 \times 128$
res2	$B(3,3) \times 2$	$26 \times 22 \times 128$
conv3	3×3 conv, stride 1, pad 0	$24 \times 20 \times 256$
pool3	2×2 conv, stride 2	$12 \times 10 \times 256$
res3	$B(3,3) \times 5$	$12 \times 10 \times 256$
conv4	3×3 conv, stride 1, pad 0	$10 \times 8 \times 512$
pool4	2×2 max-pool, stride 2	$5 \times 4 \times 512$
res4	$B(3,3) \times 3$	$5 \times 4 \times 512$
fc5	-	512
fc6	-	10575

3.2. MNIST visualization

We use a toy example similar to [30] on MNIST database to show the objective of our proposed algorithm. Specially, we define a metric Dis[3], the average cosine distance of each sample and its relevant class center, to measure the perturbations of the features, where

$$\text{Dis} = \sum_{i=1}^{N} \sum_{j=1}^{N_i} \frac{1}{N_i N} \frac{\boldsymbol{c}_i^T \boldsymbol{x}_{ij}}{\|\boldsymbol{c}_i\| \, \|\boldsymbol{x}_{ij}\|},$$

$\boldsymbol{c}_i$ is the center feature for class i, $\boldsymbol{x}_{ij}$ is the feature for class i, N_i is the feature number for class i, N is the class number.

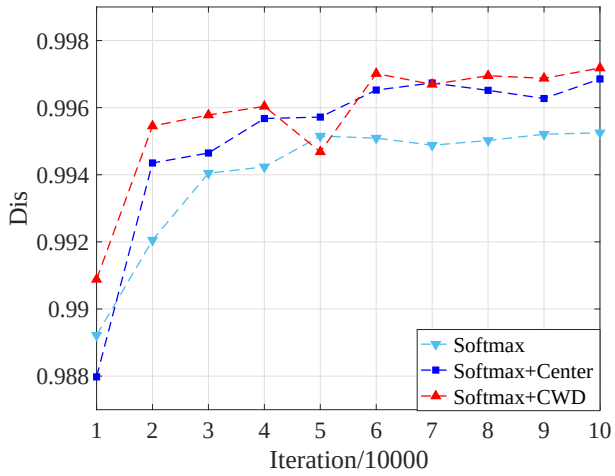
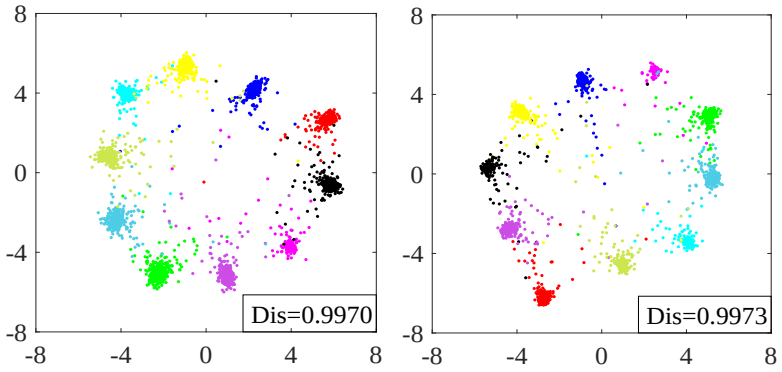


Fig. 2: We use Dis to measure the perturbations of the learned features for related approaches. For each case, as the training iteration increases, the Dis increases, which indicates that the perturbations of the features are alleviated gradually.

We record the Dis changes during the training on MNIST testing dataset in Fig. 2 and illustrate the best distributions for related approaches in Fig. 3(a) - Fig. 3(b), respectively.

[3]The larger the Dis, the less the perturbations of the features.



(a) Softmax + Center (b) Softmax + CWD

Fig. 3: Evaluation of the Dis performance for different supervision signals.

Fig. 2 shows that the Dis corresponding to our approach surpasses the other two curves by a clear margin in the end. Fig. 3 shows that the best Dis corresponding to CWD loss is larger than that of Center loss. Besides, we also observe that the diagram of a class for Softmax + CWD is slightly smaller than that in Softmax + Center. All these demonstrate the proposed approach can better alleviate the large perturbations of features during the training.

3.3. Face recognition

In this subsection, we evaluate the effectiveness of the proposed approach on ResNet-27 for face recognition, including several challenging face verification tasks and face identification tasks. The nearest neighbor and threshold comparison are used according to [30].

3.3.1. Face recognition on LFW and YTF

We choose the challenging LFW database [26] and YTF database [26] as the standard face verification benchmarks for demonstrating the effectiveness of the proposed approach. Both LFW and YTF contain the well investigated and relatively unconstrained imaging conditions, such as occlusions, poses, expressions and illuminations. The former contains $5,749$ identities of totally $13,233$ images and the latter is consists of 3,425 videos of 1,595 different identities. Besides,

we choose the public CASIA-WebFace database [36] for training CNNs, which contains $494,414$ face images amount to $10,575$ subjects.

As mentioned, CWD loss distinguishes the role of the correctly-classified features and the mis-classified features for learning. Particularly, when $\tau = 0.5$, it reduces to Center loss [30]. However, Center loss, as a special case of CWD loss, it equates the intra-class variations for the two kinds of features, which seems not enough to handle the complex perturbations during the training. To show the significance of τ, we fix the learning rate as 0.1 for total $30,000$ iterations, fix $\lambda = 0.006$ experientially according to [30], and range τ in $[0.1, 0.2, \cdots, 0.9]$ to investigate the sensitiveness. The results on LFW are shown in Fig. 4.

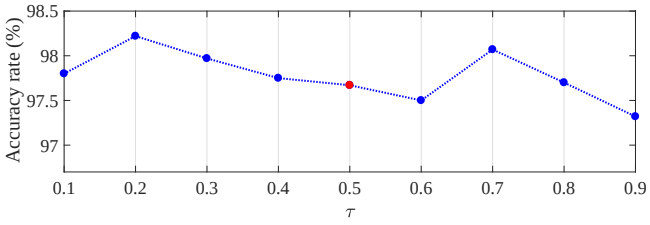


Fig. 4: Evaluation on LFW by ranging τ for ResNet-27.

From Fig. 4, the performance fluctuates as τ changes, and the best performance has achieved at $\tau = 0.2$ but not $\tau = 0.5$. That is to say, the correctly-classified features and the mis-classified features are not suitable for treating equally.

By setting the best parameter of $\tau = 0.2$ for ResNet-27, we set the initial learning rate as 0.1, then decrease it by 0.1 at $30,000$ iterations and $50,000$ iterations until reaching the maximum iteration $60,000$ to further evaluate the performance. For convenience, we denote model A, model B and model C as the final CNN models[4] supervised by Softmax loss, Softmax loss + Center loss and Softmax loss + CWD loss, respectively. Several state-of-the-art methods are also listed to compared with the proposed approach. We present the verification

[4]All CNN models used in our experiments are without fine-tuning operations. And other values of τ, that may lead to better CNN models, is beyond the scope of consideration.

results in Table 3.

Table 3: Performance (%) on LFW and YTF.

Method	#Train	LFW	YTF
FaceNet [27]	200M	99.63	95.1
DeepFace [19]	4M	97.35	91.4
VGG [42]	2.6M	98.95	97.3
NormFace [31]	1.5M	99.19	94.72
CenterApproach [30]	0.7M	99.28	94.9
WebFaceCNN [36]	0.49M	97.73	92.24
L-Softmax [32]	0.49M	98.71	-
SphereFace [33]	0.49M	99.42	95.0
DeepID2 [24]	0.2M	99.15	-
Model A	0.44M	97.82	92.66
Model B	0.44M	99.03	93.30
Model C	0.44M	**99.12**	**93.76**

The table shows that the proposed model C not only performs better than model A and model B by clear margins, but also achieves comparable performance with several state-of-the-art methods, such as DeepFace, VGG, L-Softmax and WebFaceCNN, with more less training data. These show the effectiveness of the proposed approach for learning more discriminative face features, which coincides with our analysis in Subsection 2.2 that it is necessary to give suitable treatment for different levels of perturbations, seeking a customized restraint for correctly-classified features and mis-classified features is more important than treating equally.

3.3.2. Face recognition on FGLFW

FGLFW [37] is a database 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. Thus we choose it as a more challenging image-to-image face verification benchmark. 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 4.

Table 4: Performance (%) on FGLFW.

Method	#Train	Accuracy
Noisy Softmax [43]	0.5M	94.50
CenterApproach [30]	0.7M	93.28
Human [37]	n/a	92.00
DCMN [37]	0.5M	91.00
VGG [42, 31]	2.6M	85.78
DeepFace [19, 31]	0.5M	78.78
DeepID2 [23, 31]	0.2M	78.25
Model A	0.44M	90.87
Model B	0.44M	94.28
Model C	0.44M	**95.07**

From the table, the proposed model C surpasses the baseline model B and model A by 0.79% and 4.2%, respectively. Comparing with the state-of-the-art methods in the first part of the table, model C even surpasses the second best Noisy Softmax by 0.57%. These all show that the proposed approach is effective for learning discriminative face features when it comes to more challenging face verification tasks.

3.3.3. Face recognition on BLUFR

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 of face verification tasks, with each trial containing about $156,915$ genuine matching scores and $46,960,863$ impostor matching scores on average for performance evaluation. Further, it also designs 10 random trials of face identification tasks, each trial consists about $1,000$ 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.

Table 5: Performance (%) for BLUFR protocol.

Method	TPR@FAR=0.1%	DIR@FAR=1%
NormFace [31]	95.83	77.18
CenterApproach [30, 31]	93.35	67.86
LightenedCNN [44, 31]	89.12	61.79
WebFaceCNN [36]	80.26	28.9
Model A	82.22	56.81
Model B	93.64	70.73
Model C	**94.79**	**73.69**

According to [38], we report the average TPR@FAR= 0.1% and DIR@FAR= 1% for face verification and face identification[5] in Table 5. From the table, the proposed model C surpass the baseline model B by 1.15% and 2.96% on face verification and face identification performance, respectively. Besides, it also performs better than several state-of-the-art methods on both the two face recognition tasks, such as CenterApproach, LightenedCNN and WebFaceCNN. These show that the proposed approach is also effective for learning discriminative face features for more challenging face recognition tasks.

All these experimental results demonstrate that the proposed approach, which distinguishes the role of the correctly-classified features and the mis-

[5]TAR is the true acceptance rate, FAR is the false acceptance rate, and DIR is the detection and identification rate.

classified features for restraining the intra-class variations, is an effective and easy way to learn more discriminative face representation.

3.4. Discussion

3.4.1. Limitations

The proposed approach is verified to be effective in Subsection 3.2 and Subsection 3.3. However, it still suffers from several limitations.

Firstly, the proposed approach still gives not enough treatment for the complex intra-class variations and the inter-class variations in the CNN training, which causes some poor performance. For example, the performance of the proposed approach (model C) not always better than the baseline model B, shown in Table 6, Table 7, and Table 8. And there are also many failure examples in the testing period, shown in Fig. 5.

Table 6: Number of failure examples for LFW evaluation.

Fold	1	2	3	4	5	6	7	8	9	10
Model B	9	5	5	7	11	5	5	8	0	3
Model C	8	4	2	7	7	7	7	5	2	4

Table 7: Number of failure examples for YTF evaluation.

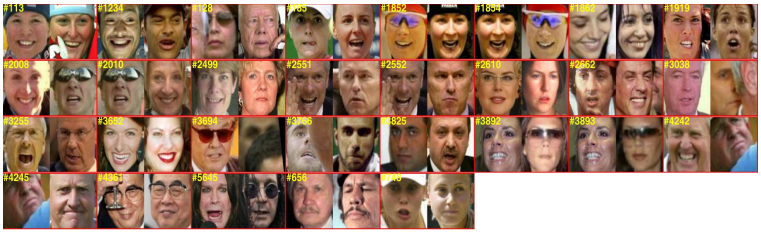
Fold	1	2	3	4	5	6	7	8	9	10
Model B	35	40	33	37	25	28	24	31	45	37
Model C	34	36	33	30	20	29	22	34	45	29

Table 8: Number of failure examples for FGLFW evaluation.

Fold	1	2	3	4	5	6	7	8	9	10
Model B	39	34	33	41	29	31	40	40	26	30
Model C	30	29	35	34	28	20	31	45	21	23

From the three tables, there are still some cases that model C performs worse than model B, which is illustrated in the number of the failure examples marked by the double underlines.

(a) All false positive matches in LFW



(b) All false negative matches in LFW

(c) All false positive matches in FGLFW of fold 8

(d) All false negative matches in FGLFW of fold 8

Fig. 5: Display of failure examples. (a) and (b) display the false positive matches and the false negative matches in the 10 folds testing of LFW, respectively. (c) and (d) display the false positive matches and the false negative matches in the 8-th fold of FGLFW, which is the most challenging of the 10 folds.

From Fig. 5, we find that the number of false negative matches are more than the number of the true positive matches in both LFW case and FGLFW case, which means that the treatment for the inter-class variations is still not enough in the proposed approach. In addition, we can also found that the false positive

matches are misclassified due to the similar facial appearances, such as the similar expression, similar pose, similar skin color, and so on. For the false negative matches, they are not only influenced by the facial appearances, but also suffer from the conditions, such as occlusion, illumination, decoration, and even the false positive detected faces, such as $\#128, \#3694, \#4242$ in LFW. For small scale database, the issue can be partly alleviated by considering more suitable preprocessing techniques, such as cropping and manual assistance. However, for more challenging video face recognition, it is intractable to deal with so many false positive faces and some undetected faces, which calls for more advanced video detection techniques, such as developing more effective video face detectors by taking advantage of [10, 14].

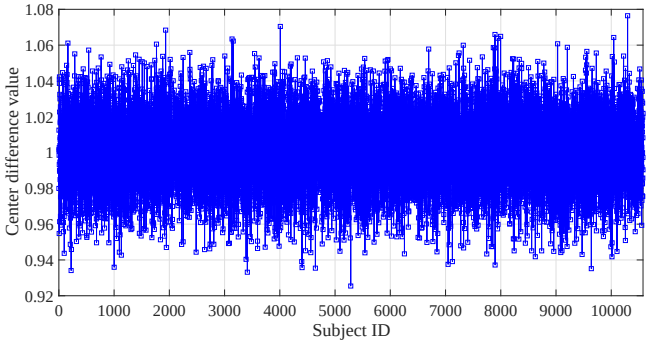
Secondly, the proposed approach does not always work for all mainstream CNN architectures. For example, we evaluate the adaptability of the proposed approach on the other two CNN architectures, AlexNet [45] and VGG-16 [42]. For AlexNet, we set the initial learning rate as 0.01, then decrease it by 0.2 every 20,000 iterations until reaching the maximum iteration 16,000, and the best hyper-parameter is $\tau = 0.8$. For VGG-16, we set the initial learning rate as 0.0001 by finetuning the model released in [42] similar to [7, 12].

Table 9: Performance (%) for AleNet.

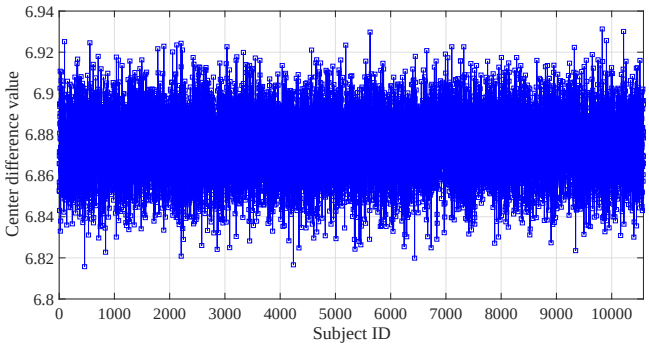
Method	#Train	LFW	YTF
Softmax	0.44M	95.32	89.84
Softmax + Center	0.44M	96.6	90.76
Softmax + CWD	0.44M	**97.42**	**91.64**

For AlexNet, Softmax + CWD gives the respectable performance, shown in Table 9. However, Softmax + CWD and Softmax + Center do not work when it comes to the VGG-16 architecture. The two strategies even causes serious divergence problems compared to simply using Softmax, which is due to the inconsistency of the initialization of the feature distribution and the

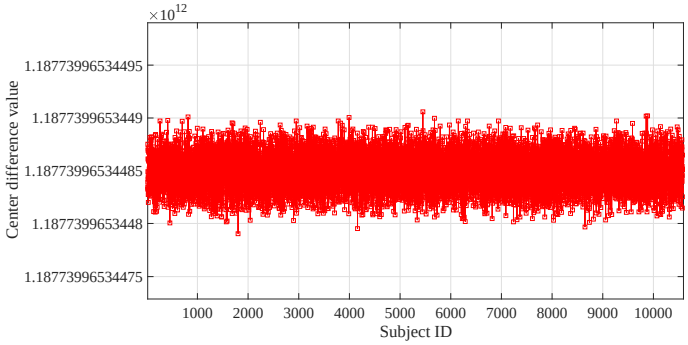
initialization of the center features[6], shown in Fig. 6.



(a) ResNet-27 case



(b) AlexNet case



(c) VGG-16 case

Fig. 6: Center difference values on WebFace databse for related CNNs.

[6]We use the same center feature initialization as is used in [30].

From the figure, the center difference value, the distance of the initial center feature and the mean of the initial features of the corresponding subject, is illustrated to measure the degree of the inconsistency for related CNN architectures. It is clear that the order of magnitude of the center difference value in VGG-16 (Fig. 6(c)) is much larger than that of the other two CNN architectures (Fig. 6(a) and Fig. 6(b)), which means that we should pay more attention to the center feature initialization to avoid such inconsistency phenomenon to further improve the proposed approach.

3.4.2. Future work

Based on the preceding discussion of the limitations of the proposed approach, the work in this paper is still insufficient and needs more in-depth study in the future. The meaningful directions are summarized in the following.

- Giving more suitable treatment for both the intra-class variations and the inter-class variations by dynamicly and effectively setting the hyper-parameter τ, and also trying cosine distance instead of L2-distance, such as [33, 31].
- Making the initialization and updating of the center feature more general for the mainstream CNN architectures, such as [46].
- Pay attention to more challenging video face recognition to dig out the potential problems existed in the proposed approach and then try to improve the performance, such as taking advantage of the merits of [10, 14].

4. Conclusion

In this paper, we propose the customized weighted discriminative (CWD) loss to learn deep discriminative face features. The aim of CWD loss is to alleviate the perturbation phenomenon by distinguishing the role of the correctly-classified features and the mis-classified features. Extensive experiments on MNIST visualization and several famous and important face recognition tasks

show the superior of the proposed approach. Detailed analysis are also reported on issues such as the limitations and future directions of the proposed appraoch.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant Number: 61801325), 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] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[2] J. Charles, T. Pfister, D. Magee, D. Hogg, A. Zisserman, Personalizing human video pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2016, pp. 3063–3072.

[3] K. Fragkiadaki, P. Arbelaez, P. Felsen, J. Malik, Learning to segment moving objects in videos, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 4083–4090.

[4] L. Wang, W. Ouyang, X. Wang, H. Lu, Visual tracking with fully convolutional networks, in: IEEE Conference on Computer Vision, 2015, pp. 3119–3127.

[5] X. Dong, J. Shen, W. Wang, Y. Liu, L. Shao, F. Porikli, Hyperparameter optimization for tracking with continuous deep q-learning, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2018, pp. 518–527.

[6] W. Wang, J. Shen, Deep visual attention prediction, IEEE Transactions on Image Processing 27 (5) (2018) 2368–2378.

[7] W. Wang, J. Shen, H. Ling, A deep network solution for attention and aesthetics aware photo cropping, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[8] W. Wang, J. Shen, Deep cropping via attention box prediction and aesthetics assessment, in: IEEE International Conference on Computer Vision, IEEE, 2017, pp. 2205–2213.

[9] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2015, pp. 3431–3440.

[10] J. Shen, J. Peng, L. Shao, Submodular trajectories for better motion segmentation in videos, IEEE Transactions on Image Processing PP (99) (2018) 1–1.

[11] W. Wang, J. Shen, X. Dong, A. Borji, Salient object detection driven by fixation prediction, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2018, pp. 1711–1720.

[12] W. Wang, J. Shen, L. Shao, Video salient object detection via fully convolutional networks, IEEE Transactions on Image Processing 27 (1) (2018) 38–49.

[13] Z. Wang, J. Ren, D. Zhang, M. Sun, J. Jiang, A deep-learning based feature hybrid framework for spatiotemporal saliency detection inside videos, Neurocomputing 287 (2018) 68–83.

[14] W. Wang, J. Shen, L. Shao, Consistent video saliency using local gradient flow optimization and global refinement., IEEE Transactions on Image Processing 24 (11) (2015) 4185–4196.

[15] B. Leng, Y. Liu, K. Yu, S. Xu, Z. Yuan, J. Qin, Cascade shallow cnn structure for face verification and identification, Neurocomputing 215 (2016) 232–240.

[16] W. Sun, H. Zhao, Z. Jin, A complementary facial representation extracting method based on deep learning, Neurocomputing 306 (2018) 246–259.

[17] Y. Li, W. Zheng, Z. Cui, T. Zhang, Face recognition based on recurrent regression neural network, Neurocomputing 297 (2018) 50–58.

[18] B. Wu, Z. Chen, J. Wang, H. Wu, Exponential discriminative metric embedding in deep learning, Neurocomputing 290 (2018) 108–120.

[19] 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 Workshops, IEEE, 2014, pp. 1701–1708.

[20] 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., Technical Report 07-49, University of Massachusetts, Amherst (2007).

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

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

[23] 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.

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

[25] D. Chen, X. Cao, L. Wang, F. Wen, J. Sun, Bayesian face revisited: A joint formulation (2012) 566–579.

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

[27] 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.

[28] O. M. Parkhi, A. Vedaldi, A. Zisserman, Deep face recognition., in: Proceedings of the British Machine Vision Conference (BMVC), Vol. 1, 2015, p. 6.

[29] 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 Conference on Computer Vision and Pattern Recognition Workshops, IEEE, 2015, pp. 142–150.

[30] 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.

[31] F. Wang, X. Xiang, J. Cheng, A. L. Yuille, Normface: L2 hypersphere embedding for face verification, in: ACM Multimedia, ACM, 2017, pp. 1141–1049.

[32] 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.

[33] 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.

[34] 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 Multimedia, ACM, 2014, pp. 675–678.

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

[36] D. Yi, Z. Lei, S. Liao, S. Z. Li, Learning face representation from scratch, arXiv preprint arXiv:1411.7923.

[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] 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.

[39] S. Wu, M. Kan, Z. He, S. Shan, X. Chen, Funnel-structured cascade for multi-view face detection with alignment-awareness, Neurocomputing 221 (2017) 138–145.

[40] J. Zhang, S. Shan, M. Kan, X. Chen, Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment, in: European Conference on Computer Vision, Springer, 2014.

[41] 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.

[42] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

[43] 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, IEEE, 2017.

[44] X. Wu, R. He, Z. Sun, A lightened CNN for deep face representation, CoRR abs/1511.02683. arXiv:1511.02683.
URL http://arxiv.org/abs/1511.02683

[45] 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.

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