Multidimensional Systems and Signal Processing (2019) 30:1517–1530
https://doi.org/10.1007/s11045-018-0614-0

A joint loss function for deep face recognition

Shanshan Wang[1] · Ying Chen[1]

Received: 29 January 2018 / Revised: 24 July 2018 / Accepted: 20 August 2018 / Published online: 24 August 2018

Abstract
Convolutional neural networks (CNNs) have been widely used in computer vision community, and significantly improving the state-of-the-art. How to train an intra-class variant and inter-class discriminative feature is a central topic in face recognition. This paper proposes to learn an effective feature from face images by a joint loss function which combines the hard sample triplet (HST) and the absolute constraint triplet (ACT) loss, under the criteria that a maximum intra-class distance should be smaller than any inter-class distance. With the joint supervision of HST and ACT loss, CNNs is enable to learn discriminative features to improve face recognition performance. Experiments on labeled faces in the wild, IARPA Janus Benchmark (IJB-A) and YouTube Faces datasets achieve a comparable or superior performance to the state-of-the-arts.

Keywords Triplet loss · Deep learning · Face recognition · Convolutional neural network · Absolute constraint

1 Introduction

Recently, deep CNNs (Convolutional Neural Networks) have boosted the FR (Face Recognition) performance to an unprecedented level. It mainly benefits from the large scale training data (Deng et al. 2009; Russakovsky et al. 2015) and the advanced network architectures (Krizhevsky et al. 2012; Simonyan and Zisserman 2014; Szegedy et al. 2015; He et al. 2016). In contrast to the conventional approaches (Guillaumin et al. 2009; Cao et al. 2010; Yin et al. 2011; Huang et al. 2012a) in face recognition, deep face recognition (Huang et al. 2012b; Cai et al. 2012; Sun et al. 2013; Liu et al. 2016; Lu et al. 2015; Wang et al. 2018; Ding and Tao 2018; Ranjan et al. 2017) typically can achieve a better performance.

Since Sun et al. (2014a) and Taigman et al. (2014) reported their work on face recognition via feature learning, most of the related work focused on how to learn effective features

✉ Ying Chen
chenying@jiangnan.edu.cn

Shanshan Wang
ivan331520@gmail.com

[1] Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, Jiangsu, People's Republic of China

from the network. Desired features are expected to be intra-class invariant and inter-class discriminative.

However, faces of the same identity could look much different when presented in different poses, illuminations, expressions, ages, and occlusions, and then caused the intrinsically large intra-class variations and high inter-class similarity that faces exhibit. Therefore, reducing the intra-class variations while enlarging the inter-class differences is a central topic in face recognition. There are two main aspects of the work in order to achieve a better performance in face recognition, focusing on the network structure construction (Simonyan and Zisserman 2014; Szegedy et al. 2015; He et al. 2016) (e.g. VGGNet, GoogLeNet and ResNet) and loss function design (Sun et al. 2014b; Schroff et al. 2015; Wen et al. 2016b).

Constructing highly efficient loss function for discriminative feature learning in CNNs is non-trivial. Softmax loss is able to directly address the classification problems. However, the softmax loss only encourage the discriminative of features. The resulting features are not sufficiently effective for face recognition. As an alternative approaches, contrastive loss (Sun et al. 2014b; Hadsell et al. 2006) and triplet loss (Schroff et al. 2015) respectively constructs loss function for image pairs and triplets. The networks of DeepID (Sun et al. 2014b) was trained by using a combination of classification and verification loss. The contrastive loss is used as verification loss and the softmax loss is used as the classification loss. However, the network structure is too complicated to implement for users.

As alternative approach, triplet loss (Schroff et al. 2015) construct loss function for triplets which include an anchor, a positive sample with the same label as anchor and a negative sample. However, non-discriminative triplet samples may be selected, where the distance of positive pair is much smaller than the negative one. Therefore, the network based on triplet loss may suffer from slow convergence and instability. By carefully selecting the image triplets, the problem may be partially alleviated. But it significantly increases the computational complexity and the training procedure becomes inconvenient.

In this paper, a joint loss function based on triplet loss is proposed which consists of a hard sample triplet (HST) which selects the triplets carefully and an absolute constraints triplet (ACT) loss which constrain the maximum intra-class distance is smaller than any inter-class distance. The loss inherits the advantage of triplet loss that aims to separate the positive pair from the negative by a distance margin in an embedding space. Meanwhile it directed against the weakness of triplet loss by imposing an absolute constraint on the loss based on the criterion that an intra-class distance should be smaller than any inter-class distance, as verified by the experiments.

Figure 1 shows the framework of the proposed algorithm. Input datas are sent to the CNN network, and distance matrix of features extracted by CNN is calculated. The features are ℓ_2-normalized and then they are sent to the proposed loss in which HST is employed to select the triplets carefully and ACT is employed to further enhance the discriminative power for learning face representations. The maximum intra-class distance and the minimum inter-class distance of one triplet are sent to the loss function.

The main contributions of this paper are summarized as follows:

- A joint loss function which consists of HST and ACT loss is proposed to satisfy the requirement that maximum intra-class instance is smaller than any inter-class instance of the deep features. With the supervision of the loss, the highly discriminative features can be obtained for robust face recognition, as supported by our experimental results.
- The proposed loss function is flexible to implement in the CNNs, and the CNN models can be directly optimized by the standard SGD (Stochastic Gradient Descent).

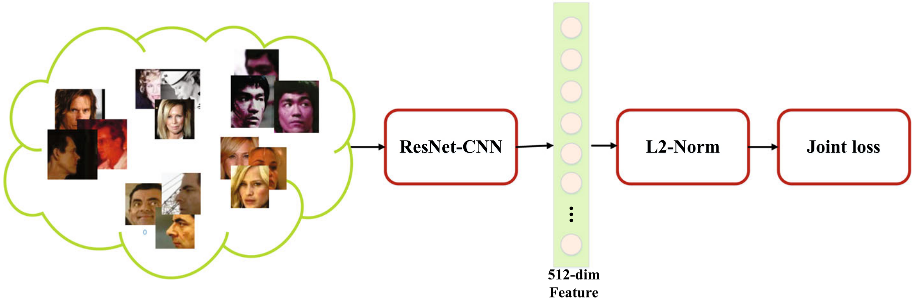

Fig. 1 Illustration of the proposed algorithm training with joint loss

- The comparable performance of our new approach is verified on Labeled Faces in the Wild (LFW) (Huang et al. 2007) and YouTube Faces (YTF) (Wolf et al. 2011) datasets.

2 Related work

There is a vast corpus of face verification and identification works. Face recognition via deep learning has achieved a series of breakthrough in these years (Taigman et al. 2014; Sun et al. 2014b; Schroff et al. 2015; Parkhi et al. 2015a; Yin et al. 2017; Wen et al. 2018). Sun et al. 2014a addressed the open-set FR using CNNs supervised by softmax loss, which essentially treats open-set FR as a multi-classification problem. Since then, people mostly focus on how to learn a discriminative feature from the deep neural network to improve the verification/identification performance.

Schroff et al. (2015) treated the FR problem using the same loss function as (Sun et al. 2014a), and it also proposed a multi-stage approach that aligns faces to a general 3D shape model. The author also experimented with a so called Siamese network where they directly optimize the ℓ_1-distance between two face features in face verification problem.

Schroff et al. (2015) used the triplet loss to learn a unified face embedding. Training on nearly 200 million face images, they achieved current state-of-the-art FR accuracy.

Sun et al. (2014a, b) proposed a compact and therefore relatively cheap method to compute network. Both PCA and a joint Bayesian model that effectively correspond to a linear transform in the embedding space were employed. The networks were trained by using a combination of classification (softmax loss) and verification loss (contrastive loss). The main difference between contrastive loss and triplet loss was that only pairs of images were compared in contrastive loss, whereas the triplet loss encouraged relative distance constraint. As can we see, most widely used loss function for deep metric learning are contrastive loss and triplet loss, and both generally impose Euclidean margin to features.

Inspired by linear discriminant analysis, Wen et al. (2016a) proposed center loss for CNNs and also obtains promising performance. There are also some loss function which improved based on the softmax loss. Liu et al. (2016, 2017a) mapped the features to angular space to obtain the discriminative features, and they have achieved excellent performance on face recognition. In this paper, an improved loss function based on the triplet loss is proposed, which is able to get a comparable result on face recognition only with a single model.

3 The joint loss

In this section, we elaborate the proposed approach. The brief review of triplet loss and introduction of the hard sample triplet (HST) loss are referred to Sect. 3.1. In Sect. 3.2, the ACT loss is presented in detail.

3.1 Brief review of the triplet loss and HST loss

Schroff et al. (2015) proposed to employ the triplet loss to train CNNs for face recognition. The representation of a face image x is ℓ_2-normalized as the input of the triplet loss. The ℓ_2-normalized face representation as $f(\mathbf{x})$ are donated. The designed representation of an anchor image $f(\mathbf{x}^a)$ of a specific subject is expected to be closer to the positive image $f(\mathbf{x}^p)$ which with the same label than the negative image $f(\mathbf{x}^n)$ with the different label. These three features$(f(\mathbf{x}^a), f(\mathbf{x}^p), f(\mathbf{x}^n))$ compose a triplet. The triplet are expected to satisfy the formula as below:

$$\|f(\mathbf{x}^a) - f(\mathbf{x}^p)\|_2^2 + \beta < \|f(\mathbf{x}^a) - f(\mathbf{x}^n)\|_2^2 \tag{1}$$

where β is the margin that satisfy the constraint between the positive pair $(f(\mathbf{x}^a), f(\mathbf{x}^p))$ and the negative pair $(f(\mathbf{x}^a), f(\mathbf{x}^n))$. The triplet loss function is formulated as below:

$$\mathbf{L}_{triplet}(f) = \frac{1}{2N}\sum_{i=1}^{N}\left[\|f(\mathbf{x}_i^a) - f(\mathbf{x}_i^p)\|_2^2 - \|f(\mathbf{x}_i^a) - f(\mathbf{x}_i^n)\|_2^2 + \beta\right]_+ \tag{2}$$

where N is the number of the triplets in a batch, and $(f(\mathbf{x}_i^a), f(\mathbf{x}_i^p), f(\mathbf{x}_i^n))$ stands for the i-th triplet. The loss is illustrated in Fig. 2a. However, non-discriminative samples may results in slow convergence and instability of the network, and the generalization of the model learned by triplet loss may be poor.

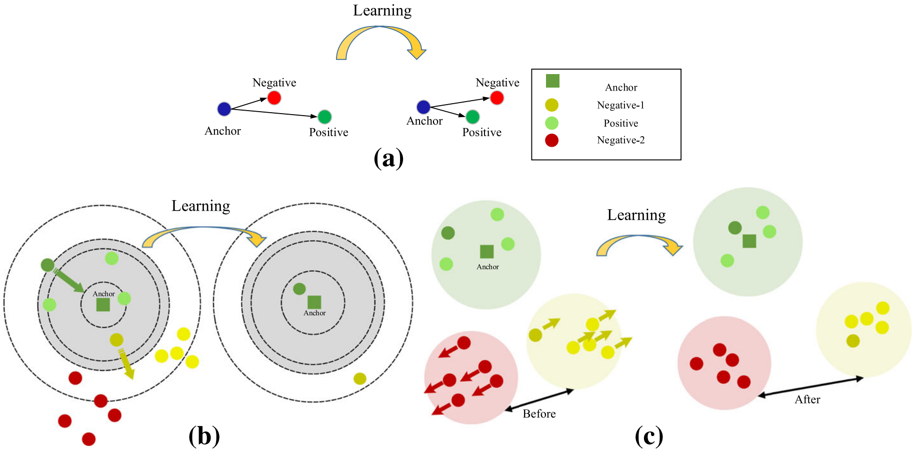

Fig. 2 Illustration of triplet, HST and ACT loss. **a** The triplet loss where the three dots with different colors stand the triplet. **b** The HST loss where the maximum intra-class distance and the minimum inter-class distance are sent to the loss function. **c** The ACT loss where any inter-class distance is required to be larger than the maximum intra-class distance (Color figure online)

The problem can be solved by using an alternative loss function called hard sample triplet (HST) loss (Hermans et al. 2017) which was proposed for person re-identification. Specifically, there are S subjects with different labels and N images for each subject in a batch. That is to say, there are S*N images in a batch or the batch size is S*N. We denote the batch set by $\chi = (f(\mathbf{x}_i), y_i)_{i=1}^{S\times N}$, where $f(\mathbf{x}_i) \in \mathbb{R}$ is the feature vector extracted from the i-th image labeled y_i. We denote the distance between any two feature vector $f(\mathbf{x}_i)$ and $f(\mathbf{x})_j$ by $d(f(\mathbf{x}_i), f(\mathbf{x}_j))$. For each positive sample pairs $(f(\mathbf{x}_i), f(\mathbf{x}_j))$ with $y_i = y_j$, we calculate their distance and compose an intra-class distance set in which the maximum intra-class distance can be selected. Similarly, for the negative sample pairs $(f(\mathbf{x}_i), f(\mathbf{x}_k))$ with $y_i \neq y_k$, we can obtain the minimum inter-class distance which is sent to the loss function, together with the maximum intra-class distance. The HST loss function is formulated as below:

$$\mathbf{L}_{HST} = \frac{1}{S \times N} \sum_{\mathbf{x}_i} \left[\max_{\mathbf{x}_j, y_i = y_j} d(f(\mathbf{x}_i), f(\mathbf{x}_j)) - \min_{\mathbf{x}_k, y_i \neq y_k} d(f(\mathbf{x}_i), f(\mathbf{x}_k)) + \alpha \right]_+ \tag{3}$$

where α is the margin that satisfy the constraint between maximum intra-class distance and minimum inter-class distance. Illustration of the HST is shown in Fig. 2b. With the help of “hard samples” satisfying the criteria that the maximum intra-class distance should be smaller than any inter-class distance, the HST loss partially alleviated the problem of slow convergence and instability occurred in the conventional triplet network.

3.2 The ACT loss

In this section, we present the proposed ACT loss as a comparison with triplet loss and HST loss.

3.2.1 Problem analysis

For triplet loss, the optimization of loss function is based on the selected triplets. However, the inter- and intra-class distances distribution are not explicit. Randomly selected triplet training samples without constraints may make the network unstable and hard to converge. It would be difficult to find an ideal threshold for face verification.

A constraint is imposed on triplet samples in HST loss in which the maximum intra-class distance and the minimum inter-class distance are sent to the loss function. As illustrated in Fig. 2b, the selected “hard sample” help to pull samples of the same identity closer while push samples of the different identity away. However, because HST considers inter- and intra-class distances relative to a certain identity when constructing a triplet, namely it considers only the relative distances of positive sample and the negative sample to a special anchor, neglecting the distance between the other negative pairs. Therefore, some negative pairs may fall into positive pairs in the distance space due to the different distance distribution of different class, as illustrated in Fig. 3 where $\mathbf{A}_i\mathbf{B}_j$ means the distance between i-th sample of class $\mathbf{A}$ and j-th sample of class $\mathbf{B}$.

3.2.2 The ACT loss

Different from the HST loss, the ACT loss imposes an absolute constraint that the maximum intra-class distance is smaller than **any** inter-class distances. In formulation, that is, for each

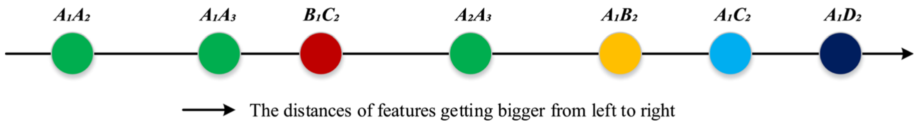

Fig. 3 Illustration of distance distribution of positive pairs and negative pairs. The green dots denote intra-class distances,and other color dots denote inter-class distances.The red dot stands the case that the negative pair fell into the positive pair (Color figure online)

batch, we aim to minimize a loss function as follows:

$$\sum_{\mathbf{x}_i}\left[\max_{\mathbf{x}_j, y_i=y_j} d(f(\mathbf{x}_i), f(\mathbf{x}_j)) - \min_{\mathbf{x}_m, \mathbf{x}_n, y_m \neq y_n} d(f(\mathbf{x}_m), f(\mathbf{x}_n)) + \beta\right]_+ \tag{4}$$

where β is a slack parameter. As illustrated in Fig. 2c, with the absolute constraint, the ACT loss pushes any two negatives apart while pull the positives close. Therefore, compared with the HST loss, it more enhances the discriminating of the learned features.

3.3 The joint loss

To hold the advantage of HST which present CNNs from slow convergence and instability, we reserve the HST loss in the algorithm. Therefore, the final loss consists of two parts. The first part is the HST loss, and the second part is the ACT loss which push the maximum intra-class distance smaller than any inter-class distance. The final loss function is formulated as follows:

$$\mathbf{L} = \mu \mathbf{L}_{HST}(\theta) + (1-\mu)\mathbf{L}_{ACT}(\theta) \tag{5}$$

where

$$\mathbf{L}_{HST} = \frac{1}{S \times N}\sum_{\mathbf{x}_i}\left(\max_{\mathbf{x}_j, y_i=y_j} d(f(\mathbf{x}_i), f(\mathbf{x}_j)) - \min_{\mathbf{x}_k, y_i \neq y_k} d(f(\mathbf{x}_i), f(\mathbf{x}_k)) + \alpha\right)_+ \tag{6}$$

$$\mathbf{L}_{ACT} = \frac{1}{S \times N}\sum_{\mathbf{x}_i}\left(\max_{\mathbf{x}_j, y_i=y_j} d(f(\mathbf{x}_i), f(\mathbf{x}_j)) - \min_{\mathbf{x}_m, \mathbf{x}_n, y_m \neq y_n} d(f(\mathbf{x}_m), f(\mathbf{x}_n)) + \beta\right)_+ \tag{7}$$

where $d(f(\mathbf{x}_m), f(\mathbf{x}_n))$ with $y_m \neq y_n$ is the distance between any two images with different labels in a batch; μ is the parameter to balance the HST loss and the ACT loss. The value of μ is same to the way of (Cheng et al. 2016) and μ is set to be 0.6; α and β are the margin force the HST loss and absolute constraint separately. We traverse the value of the margin parameter α within (0.1, 0.2, 0.3, 0.4, and 0.5), and then we choose 0.4 as the margin. Similarly, we traverse the value of the margin parameter β within (0.8, 1.0, 1.2, 1.4, and 1.6), and choose 1.2 as the margin. Equation 5 is optimized using the standard stochastic gradient descent with momentum (Jia et al. 2014).

In algorithm 1, the learning details in the CNNs with joint supervision is summarized, where $\eta(t)$ is learning rate and it starts from 0.01 and divided by 10 every 10,000 iterations.

Algorithm 1 The discriminative feature learning algorithm

```
Input:
  training set $\chi = (\mathbf{x}_i)$, $y_i$ initialized parameters $\theta$ in convolution layers, learning rate $\eta$, the number of iteration $t \leftarrow 0$.
  while not converge do
    $t \leftarrow t + 1$ samples $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k, \mathbf{x}_m, \mathbf{x}_n)$ from $\chi$
    $f(\mathbf{x}_i) = Conv(\mathbf{x}_i, \theta)$, $f(\mathbf{x}_j) = Conv(\mathbf{x}_i, \theta)$, $f(\mathbf{x}_k) = Conv(\mathbf{x}_i, \theta)$, $f(\mathbf{x}_m) = Conv(\mathbf{x}_i, \theta)$, $f(\mathbf{x}_n) = Conv(\mathbf{x}_i, \theta)$
    $\nabla f(\mathbf{x}_i) = \frac{\partial \mathbf{L}_{HST}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_k), \theta)}{\partial f(\mathbf{x}_i)} + \frac{\partial \mathbf{L}_{ACT}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_m), f(\mathbf{x}_n), \delta, \theta)}{\partial f(\mathbf{x}_i)}$, where $\delta = 1$ if $\max d(f(\mathbf{x}_i), f(\mathbf{x}_j)) > d(f(\mathbf{x}_m), f(\mathbf{x}_n))$
    $\nabla f(\mathbf{x}_j) = \frac{\partial \mathbf{L}_{HST}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_k), \theta)}{\partial f(\mathbf{x}_j)} + \frac{\partial \mathbf{L}_{ACT}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_m), f(\mathbf{x}_n), \delta, \theta)}{\partial f(\mathbf{x}_j)}$
    $\nabla f(\mathbf{x}_k) = \frac{\partial \mathbf{L}_{HST}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_k), \theta)}{\partial f(\mathbf{x}_k)} + \frac{\partial L_{ACT}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_m), f(\mathbf{x}_n), \delta, \theta)}{\partial f(\mathbf{x}_k)}$
    $\nabla f(\mathbf{x}_m) = \frac{\partial \mathbf{L}_{HST}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_k), \theta)}{\partial f(\mathbf{x}_m)} + \frac{\partial \mathbf{L}_{ACT}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_m), f(\mathbf{x}_n), \delta, \theta)}{\partial f(\mathbf{x}_m)}$
    $\nabla f(\mathbf{x}_n) = \frac{\partial \mathbf{L}_{HST}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_k), \theta)}{\partial f(\mathbf{x}_n)} + \frac{\partial \mathbf{L}_{ACT}(f(\mathbf{x}_i), f(\mathbf{x}_j), f(\mathbf{x}_m), f(\mathbf{x}_n), \delta, \theta)}{\partial f(\mathbf{x}_n)}$
    $\nabla\theta = \nabla f(\mathbf{x}_i) \cdot \frac{\partial Conv(\mathbf{x}_i, \theta)}{\partial\theta} + \nabla f(\mathbf{x}_j) \cdot \frac{\partial Conv(\mathbf{x}_j, \theta)}{\partial\theta} + \nabla f(\mathbf{x}_k) \cdot \frac{\partial Conv(\mathbf{x}_k, \theta)}{\partial\theta} + \nabla f(\mathbf{x}_m) \cdot \frac{\partial Conv(\mathbf{x}_m, \theta)}{\partial\theta} + \nabla f(\mathbf{x}_n) \cdot \frac{\partial Conv(\mathbf{x}_n, \theta)}{\partial\theta}$
    Update $\theta = \theta - \eta(t) \cdot \nabla\theta$
  end while
Output:
  The parameter $\theta$
```

4 Experiment

In Sect. 4.1, we introduced the datasets used in the experiments. The necessary implementation details are given in Sect. 4.2. In Sects. 4.3 and 4.4, extensive experiments are conducted on several public domain face datasets to verify the effectiveness of the proposed approach.

4.1 Introduction of the LFW and YTF datasets

CASIA-WebFace The CASIA-WebFace Dataset (Yi et al. 2014) is used as the training data in our experiments. The CASIA-WebFace database contains 494,414 images of 10,575 subjects. The dataset is collected with a semi-automatically way from Internet, which used under the unrestricted environment.

LFW LFW (Labeled face in the wild) (Huang et al. 2007) dataset contains 13,233 web-collected images from 5749 different identities, with large variations in pose, expression and illuminations. Following the standard protocol of unrestricted with labeled outside data.

YTF YTF (YouTube Faces) dataset (Wolf et al. 2011) consists of 3425 videos of 1595 different people, with an average of 2.15 videos per person. The clip durations vary from 48 frames to 6070 frames, with an average length of 181.3 frames.

IJB-A IJB-A (IARPA Janus Benchmark A) includes 5396 images and 20412 video frames for 500 subjects, which is a challenging with uncontrolled pose variations. Different from previous datasets, IJB-A defines face template matching where each template contains a variant amount of images. It consists of 10 folders, each of which being a different partition of the full set.

4.2 Implementation details

Preprocessing All the faces in images and their landmarks are detected by the recently proposed algorithm (Zhang et al. 2016). 5 landmarks (two eyes, nose and mouth corners) for similarity transformation are used. Finally, the faces are cropped to 112 × 96 RGB images, and each pixel in RGB images is normalized by subtracting 127.5 then dividing by 128.

Detailed settings in CNNs Caffe (2014) is used to implement ACT loss and CNNs. The ResNet-50 network is used in the experiments. For fair comparison, we respectively train four models under the supervision of softmax loss, triplet loss, HST loss and ACT loss (the latter three all used softmax for the network initialization). These models are trained with batch size of 128 with 3 blocks of parallel GPUs (1080 Ti). For the softmax loss model, the learning rate is start from 0.01, and divided by 10 every 10,000 iterations. For the next three models, it is observed that the model converges slower, and as a result, the max iteration is set 50,000.

Detailed settings in testing LFW dataset and YTF dataset, and IJB-A dataset are used to evaluate the proposed algorithm. We follow the protocol of these three datasets. For LFW dataset, there are 6000 testing pairs for the standard protocol, where 3000 of them are paired and the rest are unpaired. YTF dataset contains 10 folders of 500 video pairs. We follow the standard verification protocol and report the average accuracy on splits with cross-validation in Table 3. The deep features extracted by the network are concatenated as the representation. The score is computed by the Euclidean Distance of two features. Note that, we only use single model for all the testing.

4.3 Effectiveness of the HST loss

In this part, the HST loss on two famous face recognition benchmarks under unconstrained environments is evaluated, namely LFW and YTF datasets. They are excellent benchmarks for face recognition. Some examples of the datasets are shown in Fig. 4.

The following observations are from the results in Table 1. The HST loss beats the baseline (supervised by the softmax loss) and the triplet loss, improving the performance from (96.41% on LFW and 85.44% on YTF) to (97.25% on LFW and 87.80% on YTF) and (97.25% on LFW and 87.80% on YTF) to (98.42% on LFW and 89.86% on YTF) respectively. This shows that the HST loss can learn more discriminative features than the softmax loss and the

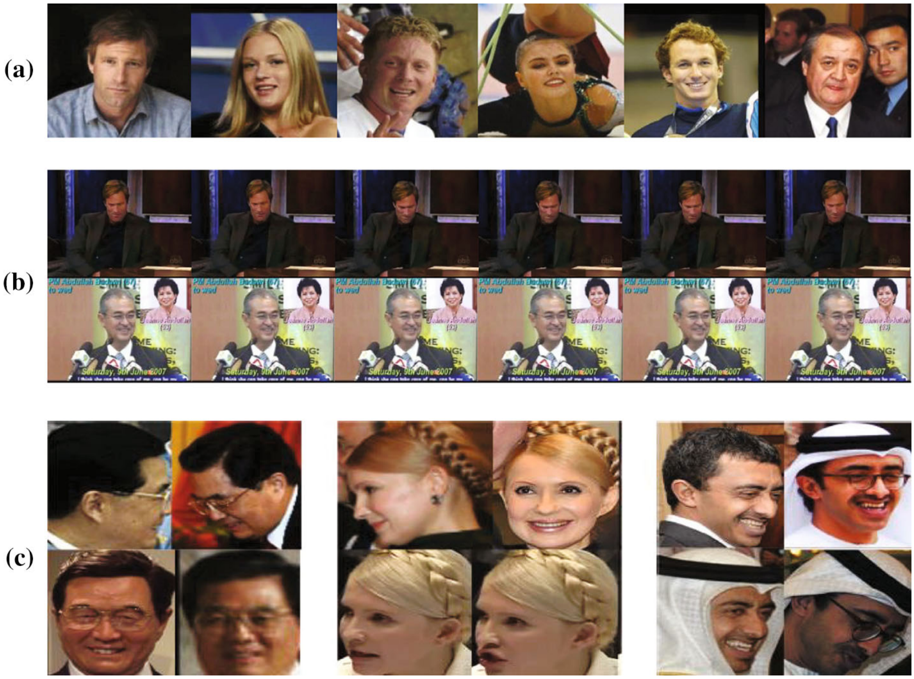

Fig. 4 **a** Examples from LFW dataset. **b** Examples from YTF dataset. **c** Examples from IJB-A dataset

Table 1 The verification rates (%) at 1% FAR (false accepted rate) of the ACT loss and the softmax loss, triplet loss and the HST loss on LFW and YFT datasets

Method	Dataset	LFW (%)	YTF (%)
Softmax	Webface	96.41	85.44
Triplet loss	Webface	97.25	87.80
HST loss	Webface	98.42	89.86
Proposed algorithm	Webface	99.23	93.14

triplet loss. Figure 5 shows the ROC curve on LFW and YTF datasets respectively, which verifies the effectiveness of the HST loss function.

4.4 Effectiveness of the ACT loss

The verification rates at 1% FAR (False Accepted Rate) of the ACT loss and the softmax loss, triplet loss and the HST loss on LFW and YTF datasets are compared in Table 1.

From Table 1, it is observed that the performance of the ACT loss on the LFW and YTF datasets are better than the softmax loss, triplet loss and the HST loss. It illustrates the feature learned by ACT loss is more discriminative than other three losses, and verifies the effectiveness of the ACT loss. Figure 5 shows the ROC curve on LFW and YTF datasets of the softmax loss, triplet loss, HST loss and ACT loss, and equally proves the effectiveness of the ACT loss.

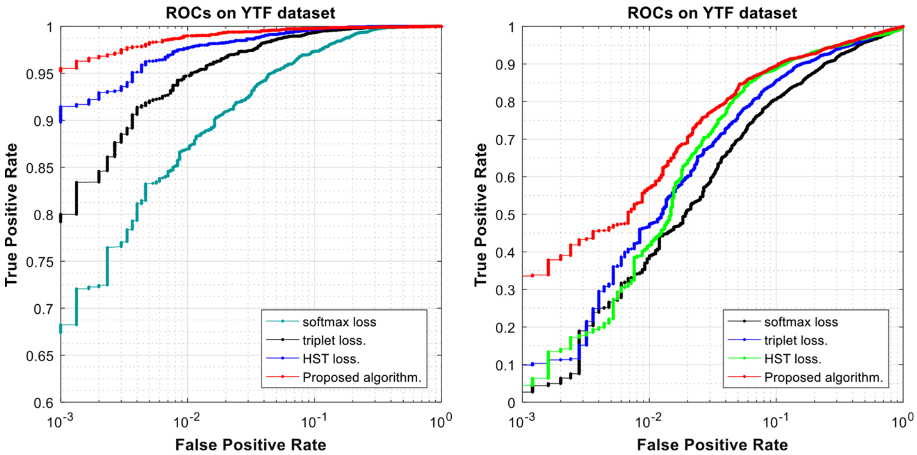

Fig. 5 The left is the ROC curves of different losses on LFW dataset, and the right one is the results on YTF dataset

4.5 Comparison with other algorithm with different loss functions

The performance of the features learned by the proposed algorithm are verified on LFW dataset, YTF dataset and IJB-A datasets, and the results are shown in Tables 2, 3 and 4 respectively.

From the results in Tables 2 and 3, one can observes that our algorithm achieve a comparable result. This shows that the ACT loss can enhance the discriminative power of deeply learned features, demonstrating the effectiveness of the ACT loss. It is worth mentioned that there only a single CNN model in our experiments, and it is easy to implement. In addition,

Table 2 The verification rates of different algorithm with different loss function on LFW dataset

Method	Dataset	LFW (%)
Deepface (Taigman et al. 2014)	4 M	97.35
Deep FR (Parkhi et al. 2015a)	2.6 M	98.95
Ding and Tao (2015)	490 K	98.43
Liu et al. (2016)	490 K	98.71
DeepID (Sun et al. 2014a)	200 K	97.45
TL Joint Bayesian (Cao et al. 2013)	100 K	96.33
GaussianFace (Lu and Tang 2015)	20 K	98.52
High-dim LBP (Chen et al. 2013)	10 K	95.17
Range loss (Zhang et al. 2017)	0.29 M	98.45
CosFace (Wang et al. 2018)	5 M	99.73
Split-Net (Wen et al. 2018)	0.49 M	99.02
ReST (Wu et al. 2017)	0.49 M	99.03
Proposed algorithm	0.49 M	99.23

Table 3 The verification rates of different algorithm on YTF dataset

Method	Dataset	LFW (%)
Deepface (Taigman et al. 2014)	4 M	91.4
LM3L (Hu et al. 2014b)	6 K	81.28
L2M3L (Hu et al. 2018)	6 K	81.72
DDML(LBP) (Hu et al. 2014a)	6 K	81.3
DDML(combined) (Hu et al. 2014a)	6 K	82.34
EigenPEP (Li et al. 2014)	9 K	84.80
VGG (Parkhi et al. 2015b)	2.6 M	91.6
DeepID2+ (Sun et al. 2015)	0.3 M	93.2
Softmax Loss (Liu et al. 2017b)	0.49 M	93.1
Proposed algorithm	0.49 M	93.14

Table 4 The verification rates of different algorithm on IJB-A dataset

Method	Dataset	IJB-A (%)
OpenBR (Klare et al. 2015)	2 M	23.6
GOTS (Klare et al. 2015)	2 M	40.6
Wang et al. (2017)	–	72.9
PAM (Masi et al. 2016)	0.49 M	73.3
DCNN (Chen et al. 2016)	0.49 M	78.7
DRGAN (Tran et al. 2017)	0.49 M	77.4
Proposed algorithm	0.49 M	75.1

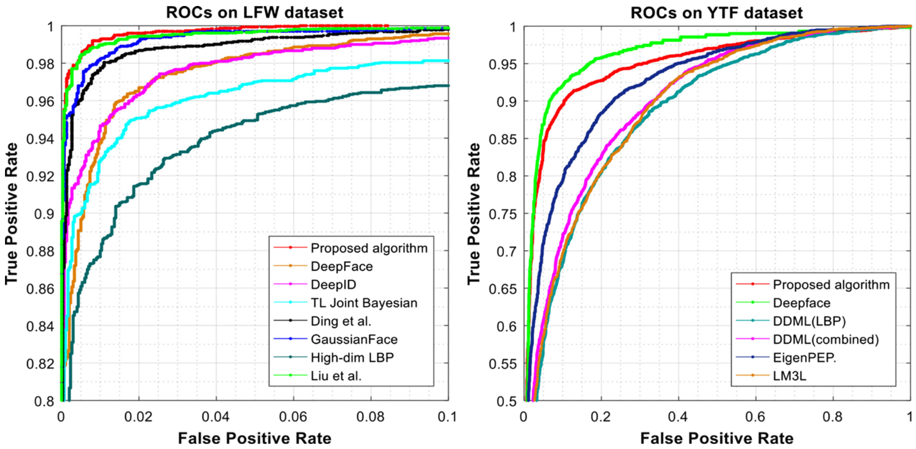

Fig. 6 The results of different methods on LFW (left) and YTF (right) dataset

the ROC curve of the different method on LFW and YTF datasets have shown in Fig. 6, and it also verifies the same statement.

5 Conclusion

In this paper, a new loss function called ACT loss is proposed. By adding an absolute constraint to the HST loss,the joint loss function make face verification difficulty caused by non-uniform of inter-class distance distribution of different identities is alleviated. The effectiveness of proposed method is verified on LFW and YTF datasets respectively. It is worth mentioned that only with a single model, it can achieve a comparable result in experiments. In addition, this work is easy to transfer to the face recognition based on videos, so our feature work may involve the recognition based on videos.

Acknowledgements This work was supported by the National Natural Science Foundation of China (61573168) and the Six Talent Summit Project talents of Jiangsu Province (2015-WLW-004).

References

Cai, X., Wang, C., Xiao, B., et al. (2012). Deep nonlinear metric learning with independent subspace analysis for face verification. In *Proceedings of the 20th ACM international conference on multimedia*. ACM.

Cao, X., Wipf, D., Wen, F., et al. (2013). A practical transfer learning algorithm for face verification. In *Proceedings of the IEEE international conference on computer vision*.

Cao, Z., Yin, Q., Tang, X., et al. (2010). Face recognition with learning-based descriptor. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2010*. IEEE.

Chen, D., Cao, X., Wen, F., et al. (2013). Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Chen, J.-C., Patel, V. M., & Chellappa, R. (2016). Unconstrained face verification using deep cnn features. In *Proceedings of the IEEE winter conference on applications of computer vision (WACV), 2016*. IEEE.

Cheng, D., Gong, Y., Zhou, S., et al. (2016). Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Deng, J., Dong, W., Socher, R., et al. (2009). Imagenet: A Large-scale hierarchical image database. In *Proceedings of the IEEE conference on computer vision and pattern recognition, 2009, CVPR 2009*. IEEE.

Ding, C., & Tao, D. (2015). Robust face recognition via multimodal deep face representation. *IEEE Transactions on Multimedia*, *17*(11), 2049–58.

Ding, C., & Tao, D. (2018). Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 1002–1014.

Guillaumin, M., Verbeek, J., & Schmid, C. (2009). Is that you? Metric learning approaches for face identification. In *Proceedings of the IEEE 12th international conference on computer vision, 2009*. IEEE.

Hadsell, R., Chopra, S., & Lecun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition, 2006*. IEEE.

He, K., Zhang, X., Ren, S., et al. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Hermans, A., Beyer, L., & Leibe, B. (2017). In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737.

Hu, J., Lu, J., & Tan, Y.-P. (2014a). Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition.*

Hu, J., Lu, J., Tan, Y.-P., Yuan, J., & Zhou, J. (2018). Local large-margin multi-metric learning for face and kinship verification. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(8), 1875–1891.

Hu, J., Lu, J., Yuan, J., et al. (2014b). Large margin multi-metric learning for face and kinship verification in the wild. In *Proceedings of the Asian conference on computer vision*. Springer.

Huang, C., Zhu, S., & Yu, K. (2012a). Large scale strongly supervised ensemble metric learning, with applications to face verification and retrieval. arXiv preprint arXiv:1212.6094.

Huang, G. B., Lee, H., & Learned-Miller, E. (2012b). Learning hierarchical representations for face verification with convolutional deep belief networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2012*. IEEE.

Huang, G. B., Ramesh, M., Berg, T., et al. (2007). *Technical report 07-49*. Amherst: University of Massachusetts.

Jia, Y., Shelhamer, E., Donahue, J., et al. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*. ACM.

Klare, B. F., Klein, B., Taborsky, E., et al. (2015). Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus Benchmark A. In *Proceedings of the IEEE conference on computer vision and pattern recognition.*

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Proceedings of the advances in neural information processing systems*.

Li, H., Hua, G., Shen, X., et al. (2014). Eigen-pep for video face recognition. In *Proceedings of the Asian conference on computer vision*. Springer.

Liu, W., Wen, Y., Yu, Z., et al. (2016). Large-margin softmax loss for convolutional neural networks. In *Proceedings of the ICML.*

Liu, W., Wen, Y., Yu, Z., et al. (2017a). SphereFace: Deep hypersphere embedding for face recognition. arXiv preprint arXiv:1704.08063.

Liu, W., Wen, Y., Yu, Z., et al. (2017b). Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR).*

Lu, C., & Tang, X. (2015). Surpassing human-level face verification performance on LFW with Gaussian face. In *Proceedings of the AAAI.*

Lu, J., Wang, G., Deng, W., et al. (2015). Multi-manifold deep metric learning for image set classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition, 2015*.

Masi, I., Rawls, S., Medioni, G., et al. (2016). Pose-aware face recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015a). Deep face recognition. In *Proceedings of the BMVC.*

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015b). Deep face recognition. In *British machine vision conference (BMVC).*

Ranjan, R., Castillo, C. D., & Chellappa, R. (2017). L2-constrained softmax loss for discriminative face verification. arXiv preprint arXiv:1703.09507.

Russakovsky, O., Deng, J., Su, H., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–52.

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

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Sun, Y., Chen, Y., Wang, X., et al. (2014b). Deep learning face representation by joint identification–verification. In *Proceedings of the advances in neural information processing systems*.

Sun, Y., Wang, X., & Tang, X. (2013). Hybrid deep learning for face verification. In *Proceedings of the IEEE international conference on computer vision*.

Sun, Y., Wang, X., & Tang, X. (2014a). Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Sun, Y., Wang, X., & Tang, X. (2015). Deeply learned face representations are sparse, selective, and robust. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2892–2900).

Szegedy, C., Liu, W., Jia, Y., et al. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Taigman, Y., Yang, M., Ranzato, M. A., et al. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Tran, L., Yin, X., & Liu, X. (2017). Disentangled representation learning gan for pose-invariant face recognition. In *Proceedings of the CVPR*.

Wang, D., Otto, C., & Jain, A. K. (2017). Face search at scale. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6), 1122–36.

Wang, H., Wang, Y., Zhou, Z., et al. (2018). CosFace: Large margin cosine loss for deep face recognition. arXiv preprint arXiv:1801.09414.

Wen, G., Mao, Y., Cai, D., & He, X. (2018). Split-Net: Improving face recognition in one forwarding operation. *Neurocomputing*. https://doi.org/10.1016/j.neucom.2018.06.030.

Wen, Y., Zhang, K., Li, Z., et al. (2016a). A discriminative feature learning approach for deep face recognition. In *Proceedings of the ECCV (7)*.

Wen, Y., Zhang, K., Li, Z., et al. (2016b). A discriminative feature learning approach for deep face recognition. In *Proceedings of the European conference on computer vision*. Springer.

Wolf, L., Hassner, T., & Maoz, I. (2011). Face recognition in unconstrained videos with matched background similarity. In *Proceedings of the IEEE Conference on computer vision and pattern recognition (CVPR), 2011*. IEEE.

Wu, W., Kan, M., Liu, X., et al. (2017). Recursive spatial transformer (rest) for alignment-free face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Yi, D., Lei, Z., Liao, S., et al. (2014). Learning face representation from scratch. arXiv preprint arXiv:1411.7923.

Yin, Q., Tang, X., & Sun, J. (2011). An associate-predict model for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2011*. IEEE.

Yin, X., Yu, X., Sohn, K., et al. (2017). Towards large-pose face frontalization in the wild. In *Proceedings of the ICCV*.

Zhang, X., Fang, Z., Wen, Y., et al. (2017). Range loss for deep face recognition with long-tailed training data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Zhang, K., Zhang, Z., Li, Z., et al. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, *23*(10), 1499–503.