J Inf Process Syst, Vol.14, No.6, pp.1464~1479, December 2018
https://doi.org/10.3745/JIPS.02.0101

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Small Sample Face Recognition Algorithm Based on Novel Siamese Network

Jianming Zhang*, Xiaokang Jin*, Yukai Liu*, Arun Kumar Sangaiah**, and Jin Wang*

Abstract

In face recognition, sometimes the number of available training samples for single category is insufficient. Therefore, the performances of models trained by convolutional neural network are not ideal. The small sample face recognition algorithm based on novel Siamese network is proposed in this paper, which doesn't need rich samples for training. The algorithm designs and realizes a new Siamese network model, SiameseFace1, which uses pairs of face images as inputs and maps them to target space so that the L_2 norm distance in target space can represent the semantic distance in input space. The mapping is represented by the neural network in supervised learning. Moreover, a more lightweight Siamese network model, SiameseFace2, is designed to reduce the network parameters without losing accuracy. We also present a new method to generate training data and expand the number of training samples for single category in AR and labeled faces in the wild (LFW) datasets, which improves the recognition accuracy of the models. Four loss functions are adopted to carry out experiments on AR and LFW datasets. The results show that the contrastive loss function combined with new Siamese network model in this paper can effectively improve the accuracy of face recognition.

Keywords

Convolutional Neural Network, Face Recognition, Loss Function, Siamese Network, Small Sample

1. Introduction

Face recognition, as a classical and important task in computer vision, is commonly used in video retrieval and pedestrian tracking, as well as distributed diagnosis and home healthcare nowadays. Stephen et al. [1] have constructed a computer model based on the cognitive learning of facial images. This computer model can both make accurate physical health judgments and predict body mass index (BMI) and blood pressure. It also facilitates the diagnosis of doctors to achieve the early identification and treatment of diseases. Meanwhile, face recognition has been extensively applied to Internet of things (IoT) such as intelligent attendance system through face identification, intelligent video surveillance system in public and secure payment system. Domestic and foreign scholars are attracted to work on face recognition because of its extensive applications [2,3]. Under the interference of factors

Manuscript received August 31, 2018; first revision October 2, 2018; accepted October 22, 2018.
Corresponding Author: Wang Jin (jinwang@csust.edu.cn)
* Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation and School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China (jmzhang@csust.edu.cn, {jxk726, lyk0311}@163.com, jinwang@csust.edu.cn)
**School of Computer Science and Engineering, Vellore Institute of Technology (VIT), Vellore, India (arunkumarsangaiah@gmail.com)

such as angle variation, illumination variation, expression and posture change, noise, low resolution, object occlusion, small number of single-class samples with numerous categories (Million), face recognition is still a challenge despite its progress. Human can recognize one person at the first sight while it is a great challenge for computer. The algorithms cope with the classifier training with small samples. The representative face recognition algorithms [4-6] indicate that the performance of these algorithms decrease dramatically with the drop of number of training samples, including the convolutional neural network (CNN) which shows excellent performance in object detection and classification [7]. Therefore, the face recognition with small sample is an extremely challenging topic.

In recent years, benefiting from the massive training data and the improvement of the hardware computing capabilities, deep learning has made great progress in the fields of image [8,9], voice [10] and text [11]. In public datasets, upon the number of category increases, the number of samples needs to be enriched to facilitate the training of network models and improve the efficiency of classification. However, there are some situations with small samples and large number of categories in real face recognition. It results in the insufficiency of samples in each category, which greatly limits the performance of face recognition.

In this work, with the help of Siamese network [12], we utilize pairs of face images as inputs to expand the number of samples for a single category, further we propose the small sample face recognition algorithm based on self-constructed Siamese network without a large amount of training samples. It presents a map by using contrastive loss functions [12] and training CNN, and maps the input image pairs to target space so that the L_2 norm distance of target space can represent the semantic distance of source space. In the training process, the network parameters learning process aims at minimizing the loss function to diminish the distance of the face image pairs from the same person and increase the distance of the face image pairs from different persons. The experiments between several loss functions are implemented and the results show that the proposed network model combined with a method to generate training data can effectively improve the face recognition accuracy, and it achieves better recognition rate on AR datasets and labeled faces in the wild (LFW) datasets.

2. Related Work

The traditional face recognition algorithms have made many achievements through years of development. The work in [13] proposes the sparse representation based classification (SRC), which uses a linear combination of all the training samples from the same person to represent one face image. The SRC compared with other ordinary methods is more effective when there are a few training samples of each category. Gabor wavelet can capture local structure information corresponding to spatial frequency, spatial position and direction. The Gabor feature is applied to SRC in [14], in which the SRC recognition rate is improved significantly. Although the SRC improves face recognition rate effectively, it causes high computational cost as well. Zhang et al. [15] propose the collaborative representation based classification (CRC), which points out that the SRC uses the regularization of the vector L_1 norm has a huge computation, while the L_2 regularization constraint can achieve similar recognition results and improves the computational efficiency as well. Nevertheless, the performances of both SRC and CRC would be greatly influenced when the number of training samples is insufficient. A new representative method called hierarchical CRC (HCRC) is proposed in [16]. Compared with some traditional collaborative representation method, HCRC introduces the Euclidean distance from

projective vectors to training vectors, which improves the recognition precision effectively even if the training sample is not enough.

In recent years, the algorithms based on CNN make great achievements in face verification and recognition [17-19]. Compared with face recognition methods based on handcraft features [20,21], CNN-based method achieves higher accuracy. A new deep learning model is proposed in [22]. It can restore the front facial features, reduce the difference between the single individual faces greatly and improve the performance of the face recognition algorithm. DeepFace [23] uses complex 3D face-lignment and four million facial images to derive a face representation from a 9-layer deep neural network. DeepID1 [24] crops the facial images, whose features are extracted from image patches and integrated by Joint Bayesian. These facial features contain rich category information. DeepID2 [25] exploits contrastive loss and softmax loss to achieve network feedback regulation. A great number of positive and negative samples are used as training data. Positive samples are used to reduce the distance of a single category. Negative samples are used to increase the distance between categories. However, the samples are generated randomly, which results in the instability of network model. A new network model called HaarNet [26] is designed. Its backbone network extracts the global image information, and its three branches use Haar-like to extract features in region of interest (ROI), which significantly improves the accuracy of the face recognition. A face recognition algorithm called FaceNet [27] maps the face images into Euclidean space, in which the distance represents the similarity of the face images. It also uses the triplet loss function in the training process, which achieves high performance in pose-variant face recognition. The number of training images is up to 200 million.

3. Face Recognition Algorithm Based on Siamese Network

According to traditional feature extraction algorithms, the feature operators are determined by hand-craft features. It is a man-made choice to extract features of a certain kind, which causes the poor robustness and expansibility of the algorithms. The advantage of CNN over traditional methods is that the parameters of the entire model are obtained by autonomous learning. It performs superiorly from following two aspects. Firstly, autonomous learning features are more robust and have stronger expressive ability. Secondly, it greatly reduces the labor and avoids that the designed parameters are not inappropriate for the model in the artificial process because of insufficient experience. CNN shows great performance in many areas of image processing and exceeds the traditional image processing methods and the human ability in some respects.

CNN can achieve great performance mainly due to the autonomous learning ability for its network model and numerous training data. CNN obtains the suitable model parameters by learning the features extracted from the training data. In conclusion, data plays a critical role in training an excellent network model. The performance of the network model would not be satisfied when we train it with a small number of data. Up to now, the recognition performance of CNN is seriously affected by insufficient face datasets of each category. In this paper, we propose the face recognition algorithm based on Siamese network. The proposed algorithm designs and implements two different network models. We can still achieve high recognition accuracy when the number of single-class training samples is small.

3.1 Siamese Network

Siamese network [12], which is divided into two parts from input to output, is one of the CNNs. Two parts of Siamese network share the same weight. Siamese network is special for that its training samples use image pairs as input, extract features by its two parts respectively, and finally obtain the eigenvector pairs of the samples. Fig. 1 shows the architecture of the Siamese network.

Here, $< \boldsymbol{X}_1 , \boldsymbol{X}_2 >$ is the input image pair. $< G_W(\boldsymbol{X}_1), G_W(\boldsymbol{X}_2) >$, calculated by network mapping, is the output feature pair. W is the parameter of the network model. $\|G_W(\boldsymbol{X}_1) - G_W(\boldsymbol{X}_1)\|_2$ is the loss function, which adjust the parameters of the entire network.

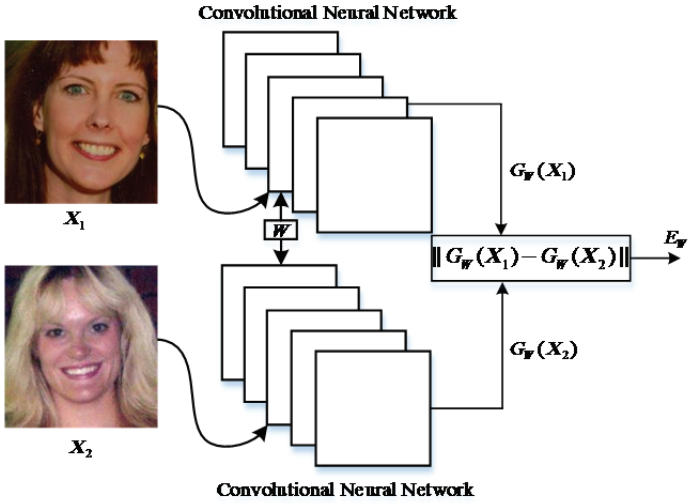

Fig. 1. Siamese network algorithm architecture.

3.2 Face Recognition Oriented Siamese Network Model Design

In this paper, we design and implement two different network models based on the Siamese network named SiameseFace1 and SiameseFace2, respectively to improve the accuracy of the face recognition.

3.2.1 SiameseFace1 model

The single network model of SiameseFace1 consists of 7 convolutional layers, 3 pooling layers, and 3 fully-connected layers. Its output is a 400-dimensional feature vector. The two outputs of the Siamese network are compared on the similarity of their Euclidean distances to judge whether they are the same type of sample. The feature pair $< G_W(\boldsymbol{X}_1), G_W(\boldsymbol{X}_2) >$ is denoted as $G_W(\boldsymbol{X}_1) = \left(x_1^{(1)}, x_1^{(2)}, \ldots x_1^{(i)}, \ldots, x_1^{(400)}\right)$ and $G_W(\boldsymbol{X}_2) = \left(x_2^{(1)}, x_2^{(2)}, \ldots x_2^{(i)}, \ldots, x_2^{(400)}\right)$ separately, the value of Euclidean distance $D < \tau$ determines that the image pair is cropped from the faces of the same person while the value of $D > \tau$ represents that the image pair is cropped from the faces of different persons. Fig. 2 shows the network architecture of SiameseFace1.

Fig. 2 shows that each input of the training are image pair and label. The label 0 denotes image pairs

from the faces of the same person while the label 1 denotes the image pairs from the faces of different persons. Sizes of the input images of the network model are set to 120×120, convolutional kernel is set to 3×3, padding is set to 1 and step is set to 1 as well. We use convolutional layers to extract features, and each convolutional layer followed by a ReLU activation function, then we employ max-pooling and three fully-connected layers. The final output of the network model is a 400-dimensional vector. Table 1 shows the detailed parameters of SiameseFace1 model.

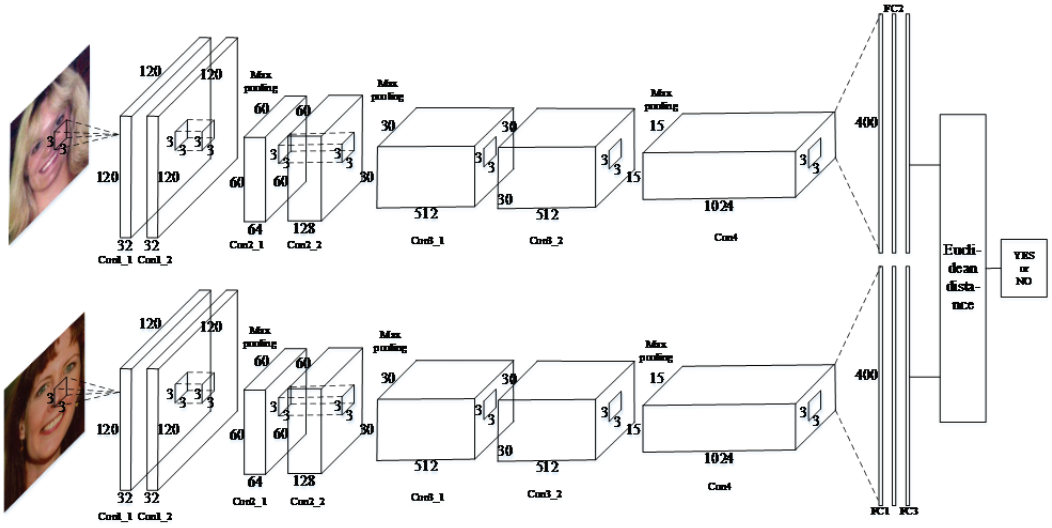

Fig. 2. SiameseFace1 network architecture.

Table 1. SiameseFace1 network parameter

Layer	Kernel	Step	Padding	Input	Output
Conv1_1	3×3	1	1	120×120×1	120×120×32
Conv1_2	3×3	1	1	120×120×32	120×120×32
Pooling	2×2	2	0	120×120×32	60×60×32
Conv2_1	3×3	1	1	60×60×32	60×60×64
Conv2_2	3×3	1	1	60×60×64	60×60×128
Pooling	2×2	2	0	60×60×128	30×30×128
Conv3_1	3×3	1	1	30×30×128	30×30×512
Conv3_2	3×3	1	1	30×30×512	30×30×512
Pooling	2×2	2	0	30×30×512	15×15×512
Conv4	3×3	1	1	15×15×512	15×15×1024
FC1				15×15×1024	400
FC2				400	400
FC3				400	400

3.2.2 SiameseFace2 model

A new lightweight network based on SiameseFace1 model is designed to optimize the network. We reduce the number of the network parameters without losing the recognition precision. SiameseFace2 network architecture is shown in Fig. 3. We add a convolutional kernel of size 1×1 in SiameseFace2

model to enhance the nonlinear eigenvalue without changing the scale of feature images. It facilitates the network deepening, enhances the feature expressive ability of the network and reduces both the dimension and the computational load at the same time. In the deep CNNs, the low-level convolutional layers extract most low-level features such as edge and texture while the high-level layers extract the features that contain more semantic information. Therefore, we cascade the low-level and high-level feature and merge detailed information (e.g., edge and texture) into semantic features in high–level layers to enhance the feature expressive ability.

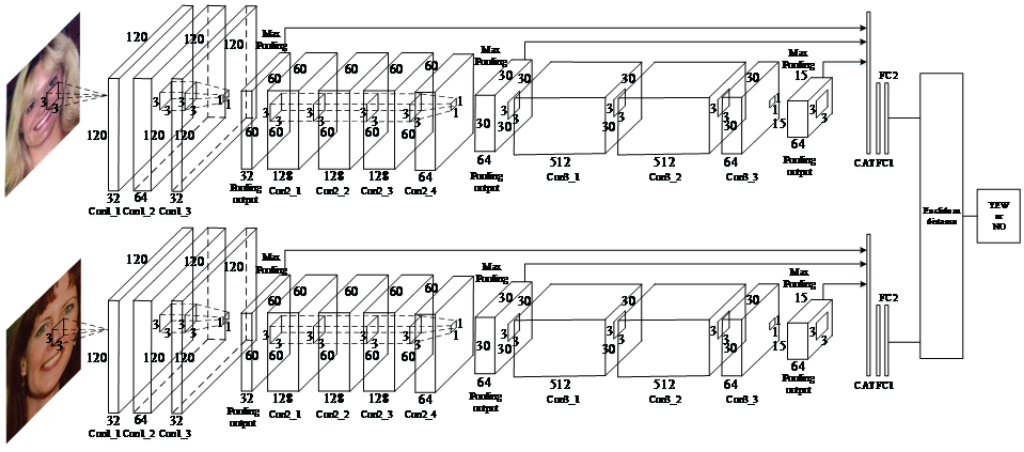

Fig. 3. SiameseFace2 network architecture.

Table 2. SiameseFace2 network parameter

Layer	Kernel	Step	Padding	Input	Output
Conv1_1	3×3	1	1	120×120×1	120×120×32
Conv1_2	3×3	1	1	120×120×32	120×120×64
Conv1_3	1×1	1	0	120×120×64	120×120×32
Pooling(out)	2×2	2	0	120×120×32	60×60×32
Conv2_1	3×3	1	1	60×60×32	60×60×128
Conv2_2	3×3	1	1	60×60×128	60×60×128
Conv2_3	3×3	1	1	60×60×128	60×60×128
Conv2_4	1×1	1	0	60×60×128	60×60×64
Pooling(out)	2×2	2	0	60×60×64	30×30×64
Conv3_1	3×3	1	1	30×30×64	30×30×512
Conv3_2	3×3	1	1	30×30×512	30×30×512
Conv3_3	1×1	1	0	30×30×512	30×30×64
Pooling	2×2	2	0	30×30×64	15×15×64
CAT	60×60×32 + 30×30×64 + 15×15×64 = 187200				
FC1				187200	100
FC2				100	100

Table 2 shows the detailed parameters of SiameseFace2 model. Out in Table 2 indicates that we cascade the feature map in this layer with CAT layer. The output of CAT is used as the input of the fully-connected layers.

3.3 Contrastive Loss Function

We use loss functions to estimate consistency between the predications $f(x)$ and ground-truth of the model. Log loss function, square loss function and exponential loss function are often used; however, they are not suitable for the Siamese network. Therefore, we employ the discriminative contrastive loss function [12] in this paper. The network parameters learning is a process of minimizing the contrastive loss function to enlarge the similarity measurement on the faces from the same person and narrow it on the faces from different persons.

As is shown in Fig. 1, $E_W^{(i)}$ denotes the Euclidean distance of the $< \boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)} >$ output features for the sample i. $E_W^{(i)}$ is computed as:

$$E_W^{(i)} = \left\| G_W(\boldsymbol{X}_1^{(i)}) - G_W(\boldsymbol{X}_2^{(i)}) \right\|_2 \tag{1}$$

We use mini-batch to process the input data in batches by CNN for more effective training. The final loss function is:

$$L(W) = \frac{1}{mb} \sum_{i=1}^{mb} H^{(i)} \tag{2}$$

Here, mb denotes the number of samples per batch. The $H^{(i)}$ represents the contrastive loss value of the sample pair $i: < \boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)} >$. The contrastive loss value $H^{(i)}$ is computed as:

$$H^{(i)} = (1 - f^{(i)}) * E_W^{(i)} + f^{(i)} * (m - E_W^{(i)}) \tag{3}$$

Here, $f^{(i)} \in \{0,1\}$ denotes the label of sample i. The label value $f^{(i)} = 0$ indicates that sample pair $i < \boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)} >$ are the faces of the same person. Its contrastive loss value $H^{(i)} = E_W^{(i)}$. The smaller $H^{(i)}$ is, the more reasonable parameters of the model are. If $H^{(i)}$ is too large, we need optimize the parameters of the model by back-propagation (BP). The label value $f^{(i)} = 1$ represents $H^{(i)} = m - E_W^{(i)}$, indicating sample pair $i < \boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)} >$ is not faces of the same person. Here, m denotes the boundary value. The similarity measurement $E_W^{(i)}$ from different faces is maximized by loss function. When $E_W^{(i)} > m$, the loss function is set to 0 without changing the model parameters. The sample pair i does not affect the network model learning process.

4. Data Training

It is difficult for the state-of-the-art face recognition algorithms to achieve models with high recognition accuracy by employing a small number of training samples for single class without pre-training. The number of the face samples for each person is relatively small in the current public face samples datasets such as AR dataset and LFW dataset. It is hard for deep learning to train an excellent network model without enough data. According to the limitations above, we reproduce the training data for the experiments based on AR and LFW dataset combined with the Siamese network.

(1) AR dataset: AR, providing 126 facial color print, is created by Purdue University in America. In this paper, we use a subset of AR dataset. This subset contains 100 person, 50 men and 50 women

respectively. Everyone has 26 images, then totally 2,600 images. The pixel for each image is 165×120. Fig. 4 shows part of the facial images of this subset.

(2) LFW dataset: In this paper, the training data originates from LFW dataset. LFW is an unconstrained face recognition dataset in scene images. The dataset consists of almost 13,000 face images of more than 5,000 celebrities in different orientations, expressions, and lighting of natural scenes. Among them, 1,680 celebrities have two or more face images per person. Each face image is a color image with size of 250×250 and has its unique name ID and serial number. Fig. 5 shows part of image dataset.

Fig. 4. AR dataset part of the face image.

Fig. 5. LFW dataset part of the face image.

The inputs of Siamese network model are image pair and label. Therefore, it is necessary to collate and generate the training data that meets the requirements. We mark the face image from the same person as 0, otherwise 1. The dataset containing 3200 pairs of images are generated at the rate of 1:1. In this dataset, 20,000 images pairs are used for training while 12,000 are used for testing. Table 3 shows the generative algorithm of the training data.

Table 3. SiameseFace2 network parameter

Training data generation algorithm	
(1)	Randomly select two face images $A_1 \in S$, $A_2 \in S$, S is the databases of face images;
(2)	If A_1 and A_2 for the same image, continue (1);
(3)	If the different images are from the same person, set label to 0; if they are from different persons, set label to 1, then, form a pair of training sample, which is T:(A_1, A_2, 0) or (A_1, A_2, 1) ;
(4)	Let the set of training samples be S', if T not exist in the sample set, added it to the set, otherwise, continue (1);
(5)	The number of training sample pair in S' reaches the set-point, end;

Fig. 6 shows the results of the matched pairs and mismatched pairs formed by the generation algorithm, where A, A_1, A_2, A_3 represent the face images of the same person with variations in expression, gesture and distinguished by different ID. Here, ***A***, ***B***, ***C***, ***D*** represent the face images of the different persons, respectively. The final sample pairs are ($A, A_1, 0$), ($A, A_2, 0$), ($A, A_3, 0$), ($A, B, 1$), ($A, C, 1$), ($A, D, 1$).

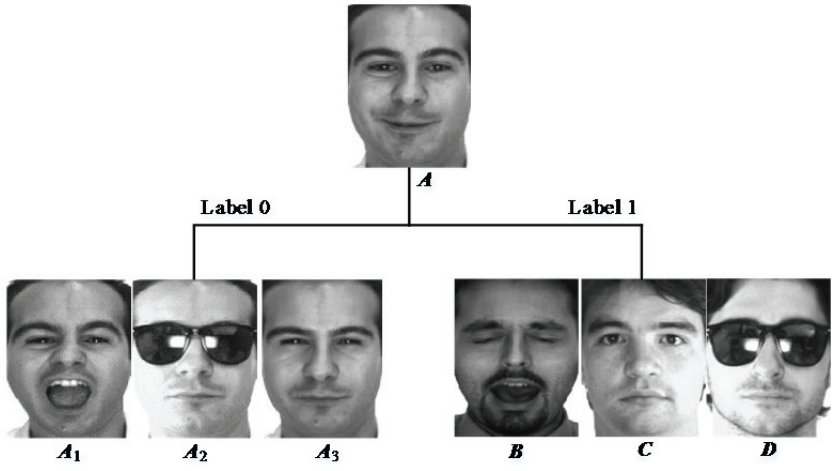

Fig. 6. Matched pairs and unmatched pairs generation.

There is no preprocessing for the images with background and illumination when features are extracted by CNN. However, *n* the process of nonlinear dimension reduction of CNN, the influence of interference factors can be eliminated automatically. To further reduce image matching time and computation, the size of image is set to 120×120.

5. Experimental Results and Analysis

The experiment is implemented on the Rongtian SCW4550 GPU server, Intel Xeon E5-2670 v3 2.3 GHz with 128 GB memory and GeForce GTX TITAN X with 12 GB memory. The processing speed is up to 50 fps which is faster than the real-time standard. We use PyTorch as the framework of deep learning. The experiment explores the effects of network structure, parameter settings and loss functions separately. We conduct our experiment on AR and LFW dataset.

We set the parameters as follows: the boundary value of the loss function m is set to 2, the mini-batch (mb) is set to 32. In an interval of [0, 30], we gradually increase the threshold with step size of 0.01. The recognition rates for each tested threshold are calculated to find the ultimate threshold which achieves the premium recognition performance. We choose 0.49 as the threshold when the highest recognition rate is up to 0.988.

Table 4. Five different network models' configuration and recognition rate

Model	model1	model2	model3	model4	model5
Network configuration	Conv3	Conv3	Conv3	Conv3	Conv3
	Conv3	Conv3	Conv3	Conv3	Conv3
	Pooling	Conv1	Conv1	Pooling	Conv1
	Conv3	Pooling (out)	Pooling	Conv3	Pooling
	Conv3	Conv3	Conv3	Conv3	Conv3
	Pooling	Conv3	Conv1	Pooling	Conv1
	Conv3	Conv3	Conv1	Conv3	Conv1
	Conv3	Conv1	Pooling	Conv3	Pooling
	Poooling	Pooling	Conv3	Pooling	Conv3
	Conv3	(out)	Conv3	Conv3	Conv3
	FC	Conv3	Conv1	Conv3	Conv1
	FC	Conv3	Pooling	Pooling	FC
	FC	Conv1	Conv3	Conv3	FC
		Pooling	Conv3	Conv3	FC
		(out)	Conv1	Pooling	
		FC	Pooling	Conv3	
		FC	Conv3	Conv3	
			Conv3	Pooling	
			Conv1 (out)	Conv3	
			Pooling	Conv3	
			Conv3	Pooling	
			Conv3		
			Conv1 (out)		
			Pooling		
			Conv3		
			Conv3		
			Conv1 (out)		
			FC		
			FC		
			FC		
Recognition rate (%)	94.8	94.6	50.1	91.1	93.2

5.1 Network Models and Parameters Comparison

In this paper, five different models are trained on LFW dataset and AR dataset, respectively. The network structure of each model is different, whose specific configuration and recognition rate are shown in Table 4.

Table 4 shows that the network structures and parameter settings will affect the accuracy of the algorithm. When the number of convolutional layers is 7, the recognition rate is the highest. When the number of convolutional layers is invariable, and the number of fully connected layers is 3, the

recognition rate is the highest. Considering comprehensively, the first model is adopted in this experiment. The size of convolutional kernel is set to 3×3 when designing network parameters, which enhances the recognition ability of the discriminant function and reduces the parameters compared with the convolutional kernel of size 5×5 and 7×7. For example, when the number of channels is C and the number of convolutional kernels of size 3×3 is 3, the number of parameters is $3\times(3\times3\times C\times C) = 27C^2$, likely, when the number of channels is C and the number of convolutional kernels of size 7×7 is 1, the number of parameters is $7\times7\times C\times C=49C^2$.

The convergence of each model is shown in Fig. 7. The model1 and model2 are corresponding to the SiameseFace1 model and the SiameseFace2 model, respectively. In the experiment, model1 has the best performance and its loss function converges fastest. Model3 is difficult to converge due to the deep network and the performance is not ideal with the same number of iterations.

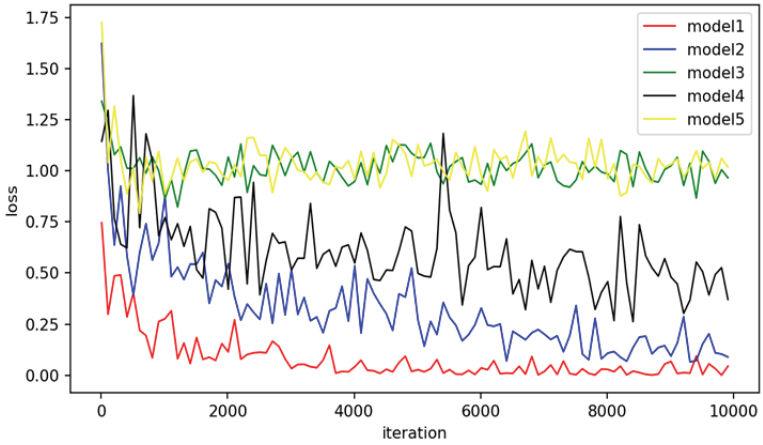

Fig. 7. Different model loss convergence performance comparison chart.

5.2 AR Dataset Experiment

AR dataset of faces contains 4,000 images of 126 people with variation in facial expression, illumination and camouflage face images. AR dataset is processed by the training data generation mode of LFW. The experimental results of the training models and the existing algorithms are shown in Table 5.

The algorithm and value marked in bold indicate that the experimental results are the best. Although the traditional algorithms in Table 5 have achieved good results on the AR dataset, the recognition rate of our algorithm has a great improvement. It shows the method based on the Siamese network can effectively solve the problem of insufficient training samples for a single category. The network model has learned effective features which can better compare and distinguish a pair of input images.

Table 5. Experimental results on the AR dataset

Algorithm	Recognition rate (%)
SRC [15]	87.9
CRC [13]	90
GSRC [14]	93
SiameseFace1	98.8
SiameseFace2	98.4

5.3 LFW Dataset Experiment

We select 12,000 pairs of faces from LFW dataset randomly to form a face test dataset, of which 6,000 pairs belong to the same person in different postures and the remaining 6,000 pairs belong to two different persons. In the test process, a pair of images are the inputs of Siamese network, the output of which is 'yes' or 'no' respectively. 'Yes' means that the image pairs represent the same person while 'no' means that the image pairs represent different person. The face recognition accuracy was obtained by the ratio of the results of 6,000 pairs of test face images to the real results. There are more than 13,000 face images collected from over 5,000 people in LFW face dataset, of which only 1,680 people have two or more images and about 4,000 people have only a face image. It greatly increases the difficulty of the model training. We only use internal data of LFW dataset when training network and don't use external data to optimize network. Table 6 shows the experimental results of the training model compared with the existing algorithms.

Table 6. Experimental results on the LFW dataset

Algorithm	Recognition rate (%)
Joint Bayesian [28]	90.90
Fisher Vector Faces [29]	93.03
FR+FCN [22]	93.65
Face++ [22]	97.27
SiameseFace1	94.80
SiameseFace2	94.60

In Table 6, the algorithm, namely Face++, is a commercial system built by a Face++ company, has the best performance. The number of facial feature points and the training data in this algorithm are not clearly opened. Our algorithm is only inferior to Face++ and has a higher recognition rate compared with other algorithms.

5.4 Comparison Experiment of Loss Function

The loss function used in this paper is the contrastive loss function, which can achieve higher recognition accuracy. We also tried to use some different loss function, including triplets-loss function, cosine proximity function, the squared error function. Comparison experiments are implemented on AR dataset and the results are shown in Table 7.

In the work of [27], the generation process of triplet is to randomly select a sample from the training dataset, denoted as *S_a*, and continue to randomly select a sample of the same class and different class with *S_a*, respectively denoted as positive samples *S_p* and negative sample *S_n*. For each element in the triple, a parameter-sharing network is trained to obtain the feature expression of the three elements, denoted as $f(s_i^a)$, $f(s_i^p)$, $f(s_i^n)$. The purpose of triplets-loss function is to make the feature expression distance between sample elements of the same class *S_a* and *S_p* as small as possible, and the distance between sample elements of different class *S_a* and *S_n* as large as possible by learning. The triplets-loss function is defined as:

$$L_{trip} = \sum_{i}^{N}\left[\| f(s_i^a) - f(s_i^p) \|_2^2 - \| f(s_i^a) - f(s_i^n) \|_2^2 + m\right]_+ \tag{4}$$

Here, N denotes the number of samples and the m denotes the margin value, subscript + represents the value in the brackets is the loss value when it is greater than zero. When it less than zero, the loss is zero.

The cosine distance is called cosine similarity in [30], which uses the cosine values of two vector angles in vector space to measure the difference between two inputs. It is defined as:

$$L_{\cos} = \frac{1}{mb}\sum_{i=1}^{mb}[(1-f^{(i)}) * \cos(\boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)}) + f^{(i)}(m + cos(\boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)})] \tag{5}$$

The loss function used in [31] is the squared error loss function, which is defined as:

$$L_{sq} = \frac{1}{mb}\sum_{i=1}^{mb}\left[(1-f^{(i)}) * \left(\frac{1}{2} - \delta\left((d^{(i)})^2\right)\right)^2 + f^{(i)} * \left(1 - \delta\left((d^{(i)})^2\right)\right)^2\right] \tag{6}$$

$$\delta(x) = \frac{1}{1+e^{-x}} \tag{7}$$

Here, δ is a logistic function, $d^{(i)}$ represents the similarity measure of sample pair $i :< \boldsymbol{X}_1^{(i)}, \boldsymbol{X}_2^{(i)} >$. A shortcoming of the squared error loss function is easy to vanish gradient.

Table 7. Comparison of different loss functions on AR dataset

Loss function	Recognition rate (%)
Triplets-loss function	98.5
Cosine proximity function	97.2
The squared error function	96.5
Contrastive loss function	98.8

As show in Table 7, the contrastive loss function used in this paper is optimal, and its recognition rate is much higher than that of the cosine proximity function, slightly higher than the triplets-loss function. It is also found that the triplets-loss function is slow and prone to overfit in the experiment.

6. Conclusion

In this paper, we propose an effective face recognition algorithm based on a novel Siamese CNN, which indirectly expands the number of training samples of a single category on AR and LFW datasets. With the image pair as the input of the network, the designed Siamese network model is used to extract the features, and the similarity calculation is carried out by using the contrastive loss function. In addition, a lightweight network model without loss of recognition accuracy is also proposed. The training data generation method combined with the new Siamese network model proposed, and the contrastive loss function, achieve a higher recognition rate on the AR and LFW datasets. In the future, we will carry out quantitative experiment analysis for single sample training, design and optimize the

deep network model, construct novel loss function and further improve the recognition performance of our algorithm.

Acknowledgement

The research work was supported by National Natural Science Foundation of China (No. 61772454, 61811530332), the Scientific Research Fund of Hunan Provincial Education Department (No. 16A008), the Scientific Research Fund of Hunan Provincial Transportation Department (No. 201446), the Industry-University Cooperation and Collaborative Education Project of Department of Higher Education of Ministry of Education (No. 201702137008), the Undergraduate Inquiry Learning and Innovative Experimental Fund of CSUST (No. 2018-6-119), and the Postgraduate Course Construction Fund of CSUST (No. KC201611).

References

[1] I. D. Stephen, V. Hiew, V. Coetzee, B. P. Tiddeman, and D. I. Perrett, "Facial shape analysis identifies valid cues to aspects of physiological health in Caucasian, Asian, and African populations," *Frontiers in Psychology*, vol. 8, article no. 1883, 2017.

[2] R. Blanco-Gonzalo, N. Poh, R. Wong, and R. Sanchez-Reillo, "Time evolution of face recognition in accessible scenarios," *Human-centric Computing and Information Sciences*, vol. 5, article no. 24, 2015.

[3] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, et al., "IARPA Janus Benchmark-C: face dataset and protocol," in Proceedings of the 11th IAPR International Conference on Biometrics (ICB), Gold Coast, Australia, 2018.

[4] F. Liu, Y. Bi, Y. Cui, and Z. Tang, "Local similarity based linear discriminant analysis for face recognition with single sample per person," in *Computer Vision-ACCV 2014 Workshop*. Cham: Springer, 2014, pp. 85-95.

[5] F. Tsalakanidou, D. Tzovaras, and M. G. Strintzis, "Use of depth and colour eigenfaces for face recognition," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1427-1435, 2003.

[6] X. He, S. Yan, Y. Hu, P. Niyogi, and H. J. Zhang, "Face recognition using laplacianfaces," *IEEE Transactions on Pattern Analysis And Machine Intelligence*, vol. 27, no. 3, pp. 328-340, 2005.

[7] Y. Tu, Y. Lin, J. Wang, and J. U. Kim, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *Computers Materials & Continua*, vol. 55, no. 2, pp. 243-254, 2018.

[8] N. Yu, Z. Yu, F. Gu, T. Li, X. Tian, and Y. Pan, "Deep learning in genomic and medical image data analysis: challenges and approaches," *Journal of Information Processing Systems*, vol. 13, no. 2, pp. 204-214, 2017.

[9] K. M. Koo and E. Y. Cha, "Image recognition performance enhancements using image normalization," *Human-centric Computing and Information Sciences*, vol. 7, article no. 33, 2017.

[10] T. N. Sainath, R. J. Weiss, K. W. Wilson, B. Li, A. Narayanan, E. Variani, et al., "Multichannel signal processing with deep neural networks for automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 965-979, 2017.

[11] S. G. Lee, Y. Sung, Y. G. Kim, and E. Y. Cha, "Variations of AlexNet and GoogLeNet to improve Korean character recognition performance," *Journal of Information Processing Systems*, vol. 14, no. 1, pp. 205-217, 2018.

[12] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, CA, 2005, pp. 539-546.

[13] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210-227, 2009.

[14] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with Gabor occlusion dictionary," in *Computer Vision-ECCV 2010.* Heidelberg: Springer, 2010, pp. 448-461.

[15] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: which helps face recognition?," in *Proceedings of 2011 IEEE International Conference on Computer Vision (ICCV),* Barcelona, Spain, 2011, pp. 471-478.

[16] D. M. Vo and S. W. Lee, "Robust face recognition via hierarchical collaborative representation," *Information Sciences*, vol. 432, pp. 332-346, 2018.

[17] C. Li, S. Zhao, K. Xiao, and Y. Wang, "Face recognition based on the combination of enhanced local texture feature and DBN under complex illumination conditions," *Journal of Information Processing Systems*, vol. 14, no. 1, pp. 191-214, 2018.

[18] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. C. Adams, T. Miller, et al., "IARPA Janus Benchmark-B face dataset," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Honolulu, HI, 2017, pp. 90-98.

[19] L. Tran, X. Yin, and X. Liu, "Disentangled representation learning GAN for pose-invariant face recognition," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp. 1415-1424.

[20] D. Chen, X. Cao, F. Wen, and J. Sun, "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*, Portland, OR, 2013, pp. 3025-3032).

[21] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher vector faces in the Wild," in *Proceedings of the British Machine Vision Conference (BMVC)*, Bristol, UK, 2013.

[22] Z. Zhu, P. Luo, X. Wang, and X. Tang, "Recover canonical-view faces in the wild with deep neural networks," 2014 [Online]. Available: https://arxiv.org/abs/1404.3543.

[23] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf, "DeepFace: closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 1701-1708.

[24] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 1891-1898.

[25] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," *Advances in Neural Information Processing Systems*, vol. 27, pp. 1988-1996, 2014.

[26] M. Parchami, S. Bashbaghi, and E. Granger, "Video-based face recognition using ensemble of Haar-like deep convolutional neural networks," in *Proceedings of 2017 International Joint Conference on Neural Networks (IJCNN),* Anchorage, AK, 2017, pp. 4625-4632.

[27] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: a unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 815-823.

[28] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: a joint formulation," in *Computer Vision-ECCV 2012.* Heidelberg: Springer, 2012, pp. 566-579.

[29] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz, "Fast high dimensional vector multiplication face recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, Sydney, Australia, 2013, pp. 1960-1967.

[30] S. Berlemont, G. Lefebvre, S. Duffner, and C. Garcia, "Class-balanced Siamese neural networks," *Neurocomputing*, vol. 273, pp. 47-56, 2018.

[31] U. Shaham and R. R. Lederman, "Learning by coincidence: Siamese networks and common variable learning," *Pattern Recognition*, vol. 74, pp. 52-63, 2018.

Jianming Zhang https://orcid.org/0000-0002-4278-0805

He received the B.S. and M.S. degree in 1996 and 2001, respectively from Zhejiang University and the National University of Defense Technology, China. He received the Ph.D. in 2010 from Hunan University, China. Currently, he is an associate professor and the deputy dean in the School of Computer and Communication Engineering at Changsha University of Science and Technology, China. His main research interests lie in the areas of computer vision, data mining, and wireless ad hoc & sensor networks.

Xiaokang Jin https://orcid.org/0000-0002-9563-8888

He received the B.S. degree from the Changsha University of Science and Technology in 2016, China. He is currently pursuing the M.S. degree in computer science and technology at Changsha University of Science and Technology. His research interests include computer vision, deep learning and object tracking

Yukai Liu https://orcid.org/0000-0003-3263-7543

He received the B.S. degree in 2013 from Xiangnan University, China. He received the M.S. degree from Changsha University of Science and Technology in 2018, China. He is now a computer vision algorithm engineer of Hunan SHUDING Intelligent Technology Co., Ltd. His research interests include computer vision, deep learning and pattern recognition.

Arun Kumar Sangaiah https://orcid.org/0000-0002-0229-2460

He received the M.S. degree in computer science and engineering from the Government College of Engineering, Tirunelveli, Anna University, India. He received the Ph.D. degree in computer science and engineering from the VIT University, Vellore, India. He is presently working as an associate professor in the School of Computer Science and Engineering, VIT University, India. His area of interest includes software engineering, computational intelligence, wireless networks, bioinformatics, and embedded systems.

Jin Wang https://orcid.org/0000-0002-6516-6787

He received the B.S. and M.S. degrees from Nanjing University of Posts and Telecommunications, China in 2002 and 2005, respectively. He received Ph.D. degree from Kyung Hee University, Korea in 2010. Now, he is a professor in the School of Computer & Communication Engineering, Changsha University of Science and Technology. His research interests mainly include wireless communications and networking, performance evaluation and optimization, etc.