Information Sciences 432 (2018) 332–346

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Robust face recognition via hierarchical collaborative representation

Duc My Vo, Sang-Woong Lee*

Pattern Recognition and Machine Learning Lab, Gachon University, 1342 Seongnamdaero, Sujeonggu, Seongnam 13120, Republic of Korea

ARTICLE INFO

Article history:
Received 29 March 2017
Revised 4 December 2017
Accepted 7 December 2017
Available online 11 December 2017

Keywords:
Face recognition
Hierarchical collaborative representation-based classification
Local ternary patterns
Convolutional neural network

ABSTRACT

Collaborative representation-based classification (CRC) is currently attracting the attention of researchers because it is more effective than conventional representation-based classifiers in recognition tasks. CRC has shown high face recognition accuracy; however, its accuracy is degraded significantly if the number of training faces in each class is small. This is because the accuracy of CRC is only dependent on the results of minimizing the Euclidean distance between a testing face and its approximator in the collaborative subspace of training faces. In this research, we proved that the accuracy of CRC can be improved substantially by minimizing not only the Euclidean distance between a testing face and its approximator but also the Euclidean distances from the approximator to training faces in each class. Consequently, we presented a hierarchical collaborative representation-based classification (HCRC) in which a two-stage classifier is applied for training faces, and the recognition accuracy of the second-stage classifier is significantly improved in comparison to that of the first-stage classifier. Moreover, the recognition rate of our classifier can be considerably increased by using models of discriminative feature extraction. Since noise and illumination are the main factors that cause CRC to be less accurate, we propose combining HCRC with a wide model of local ternary patterns (LTP). This combination enhances the efficiency of face recognition under different illumination and noisy conditions. For dealing with face recognition under variations in pose, expression and illumination, we present a deep convolutional neural network (DCNN) model of discriminative feature learning, which transforms face images into a common set of distinct features. The combination of HCRC with this deep model achieves high recognition rates on challenging face databases. Furthermore both models are optimized to reduce computational costs so that they can be successfully applied for real-world applications of face recognition that are required to run reliably in real time. In addition, we also prove that combining state-of-the-art DCNN models with HCRC results in an significant improvement in face recognition performance. We demonstrate several experiments with challenging face recognition datasets. Our results show that the hierarchical collaborative representation-based classifier with the models significantly outperforms state-of-the-art methods.

* Corresponding author.
E-mail address: slee@gachon.ac.kr (S.-W. Lee).

https://doi.org/10.1016/j.ins.2017.12.014
0020-0255/© 2017 Elsevier Inc. All rights reserved.

1. Introduction

In the last two decades, the field of face recognition has made noteworthy contributions to the development of security systems and mobile robots. The tool of face recognition improves the ability of systems to perform difficult tasks that previously might have required a large amount of human effort, for example, security management for apartments, airports, and agencies. Such a system requires to run reliably in real time, which presents greater challenges than conventional recognition systems. A conventional face recognition system may achieve a high accuracy rate but its computational costs are also very high. Fortunately, the widespread development of high performance accelerators for data parallel computing has been a key advantage to the success of recent research in classification algorithms, for instance, face recognition using deep learning, which motivated us to find better solutions for a fast face recognition system. For this reason, in this paper, we propose a new face recognition algorithm to quickly and reliably identify individuals. This algorithm is applicable to real-time face recognition systems with the support of graphics processing units (GPUs).

The limited number of available training faces is one of the major challenges of face recognition, because collecting high number of training faces is not an easy task in applications for security systems or mobile robots. Thus, sparse representation-based classification (SRC) [26] has recently attracted the attention of researchers, because SRC can represent a facial feature vector as a linear combination of the training vectors on the entire dataset instead of each subset. SRC is more effective than conventional algorithms in recognition tasks using a small number of training faces in each class. SRC has shown a high face recognition accuracy rate; however, its computational cost is expensive. Recently collaborative representation-based classification (CRC) [29] has been shown to be a better solution than SRC, and it is not only as accurate as SRC, but also much less time-consuming. However, we found one of the main drawbacks of this representation needed to be solved, that is the problem posed by the limited number of available training faces. Focusing on resolving this primary problem, we propose a new representation called hierarchical collaborative representation-based classification (HCRC) to address effectively the problem of an inadequate number of training samples. Our algorithm achieves a higher accuracy rate than CRC and state-of-the-art algorithms on challenging datasets. The reason for this increased accuracy is that the original collaborative representation focuses only on minimizing the Euclidean distance from the testing face to its projection vector in the collaborative subspace of training faces, while the Euclidean distances from this projection vector to training vectors are not considered. Unlike CRC, HCRC takes both distances into account to improve the recognition rate.

In our research, we also found that noise and illumination are additional significant factors that render the collaborative representation-based algorithm considerably less effective in realistic environments. In order to solve these main problems, we focused on building feature extraction models to generate discriminative features that are insensitive to noise and illumination. We present two feature extraction models that are useful for different realistic applications of face recognition. The first is a DCNN model, which is applicable to extracting distinctive features. We designed a maxout network for real-time applications of face recognition. In addition, we also combined state-of-the-art DCNN models with HCRC to improve recognition accuracy. The second is a LTP model, which was proved to be insensitive to random noise and significantly reduce the effect of uncontrolled illumination.

The first stage of the deep feature learning method is to build a deep model that transforms face images into a common set of distinct features. A model with a very deep architecture can be more successful for training faces than traditional models with the same number of parameters. Further, DCNNs have recently become the state-of-the-art deep models for unconstrained face recognition. Since a DCNN model is trained on a very large training dataset, almost all the discriminative facial features are collected to build a deep face recognition model. However, building a very deep face recognition model is expensive because of the huge number of training parameters required. In this study, we not only built a very deep convolutional network to train the best facial features but also minimized the number of training parameters so that it can run in real time with the support of GPUs.

Descriptor-based algorithms such as local binary patterns (LBP) [1] and LTP [23] are promising approaches for extracting features of shape and texture to improve the face recognition accuracy rate. The key advantage of these descriptors is that they are invariant to gray-scale changes and are much less time consuming than other descriptors. Furthermore, LTP even outperforms LBP in dealing with difficult illumination conditions and noise. Thus, they have achieved a considerable success in uncontrolled face recognition. For this reason, to handle the problems of illumination and noise, we propose combining an LTP extraction model with HCRC. This combination enhances the efficiency of the HCRC in dealing with noise and illumination.

We tested our proposed algorithms and its competitors to evaluate the accuracy of face recognition in different environments. Based on extensive experiments, our algorithms are shown to significantly outperform state-of-the-art algorithms. In particular, our contributions are summarized as follows:

- We presented a new HCRC algorithm that significantly outperforms competing algorithms.
- We proposed an algorithm for recognizing faces, based on the combination of a LTP model and HCRC. This combination leads to a better performance when noise and challenging illumination conditions have to be addressed.
- We trained a maxout network for extracting face features. We then developed an approach in which this deep model of discriminative feature learning is combined with HCRC. This algorithm not only achieved a high face recognition accuracy rate on challenging datasets but is also a promising algorithm for real-time applications of face recognition.

- HCRC was also combined with a very deep convolutional networks (VGG) [15] to address challenges in face recognition. This approach is the best among competitive approaches using deep learning models.
- We demonstrated that state-of-the-art face recognition algorithms using DCNN models have not fully addressed the challenging problem of recognize human faces from seriously noisy images. Their recognition accuracy rates drop dramatically when random noise increases in testing images.

The remaining parts of this paper are organized as follows. In Section 2, we briefly review some related state-of-the-art face recognition algorithms as well as feature learning models, which motivated our research. Section 3 describes the CRC method. In Section 4, we present in detail our proposed classifier for face recognition. Section 5 presents our facial feature extraction models. In Section 6, the experimental results obtained from some challenging face databases are presented. We conclude this paper by describing our intentions with regard to our future work, in Section VII.

2. Related work

Recently, sparse representation, developed from the theory of sparse coding, is drawing the attention of researchers in the fields of pattern recognition, object detection, and especially face recognition [4]. In this type of representation, a testing face is represented as a combination of all the training faces on the training dataset instead of by each subspace. Then, this face is classified based on the least representation residual. In the latest research on sparse representation, Imran [14] proposed a linear model (LRC) in which a testing face can be represented as a linear combination of class-specific galleries, and the least-squares method is applied to resolve its inverse problem. Some advanced versions of sparse representation are used to resolve misalignment and pose change problems. The collaborative representation that uses non-sparse L_2-regularization instead of L_1-norm sparse regularization was proposed in [29]. This improvement made a significant difference between CRC and SRC. CRC is almost as accurate as SRC while it is much less time-consuming. Motivated by CRC, researchers have recently developed new face recognition algorithms and they have achieved significant progress in improving recognition rates. Liu [11] explored kernel techniques (KCRC) to transform nonlinear data into high dimensional feature spaces so that training data is more separable. The new features in kernel space can be trained by CRC to gain a better recognition performance. Since it is difficult to collect a large number of training faces in realistic applications of face recognition, Zhu [30] proposed a multi-scale patch-based CRC method (MSPCRC) for solving the challenging problem posed by a small sample size and improving the performance of the original CRC. By combining the information on different scales, MSPCRC leads to remarkable improvement in the face recognition rate. Furthermore, in each scale, Zhu proposed to use a patch-based CRC in which the testing image is classified by combining the recognition outputs of overlapped patches. This method motivated us to develop a recognition method combining global and local features, which can provide complementary information to improve recognition rates. Kumar [10] also segmented the face image into specific sub blocks and built a functional mapping using truncated Volterra kernels. This approach (Volterra) can extract discriminative information from the null space of the intra-class. Yang [28] divided the face image into multiple blocks, and stretched each block to a feature vector. He proposed a relaxed collaborative representation (RCR) model to compute both the similarity and distinctiveness of feature vectors in training and testing stages. Recent research has also employed multiple types of features for joint sparse representation and recognition. Wright [26] and Yang [27] improved SRC to address the challenging problems of face recognition, including illumination changes, random pixel corruption, block occlusion and real disguise, etc.

Recently, deep neural networks have also been used in pattern classification and feature extraction [3,20–22]. By training features on very large datasets, several deep learning methods achieved very high face recognition rates. The convolutional neural network (CNN) is still the state-of-the-art deep model for face recognition. DCNNs are usually designed to build a deep model that automatically transforms input images into a common set of distinct features. This model with a very deep architecture can be more successful for training than traditional models with the same number of parameters. Therefore, many excellent methods of training DCNNs are successful in learning face representations from very large datasets. Although these methods achieved remarkable recognition rates of between 97% and 99%, running the face recognition deep models are extremely time-consuming because of a huge number of training parameters. In order to reduce computational costs, Goodfellow [7] developed a maxout network which plays an important role in minimizing the number of necessary neurons and the number of network parameters in each layer. A maxout layer is also considered a set of efficient activation functions that are faster than traditional activation functions. Schroff [18] successful applied maxout networks to gain high face recognition accuracy rates on challenging datasets. Some DCNNs even get close to human performance when testing on the LFW dataset [9]. The CenterLoss model [24] also aims to learn a center for deep features of each class and penalize the distances between the deep features and those of their corresponding class centers. As a result, intra-personal variations are reduced and inter-personal differences are enlarged. This model achieved the state-of-the-art face recognition accuracy rate on the MegaFace Challenge database [19]. The VGG model [15] exploits the effect of the convolutional network depth on its recognition accuracy rate. It was built based on a very deep architecture of convolution layers with very small convolution filters. It has been ranked in the top in general image classification in the ImageNet large-scale visual recognition challenge (ILSVRC) 2014 [16].

It is a fact that a conventional system of face recognition is usually vulnerable to noise during its acquisition, quantization and compression. Furthermore, it is even difficult to recognize human faces from a seriously noisy image by human. Some good approaches used the fuzzy edge detectors [6,13,17] that provide distinctive features of human faces and can be used to

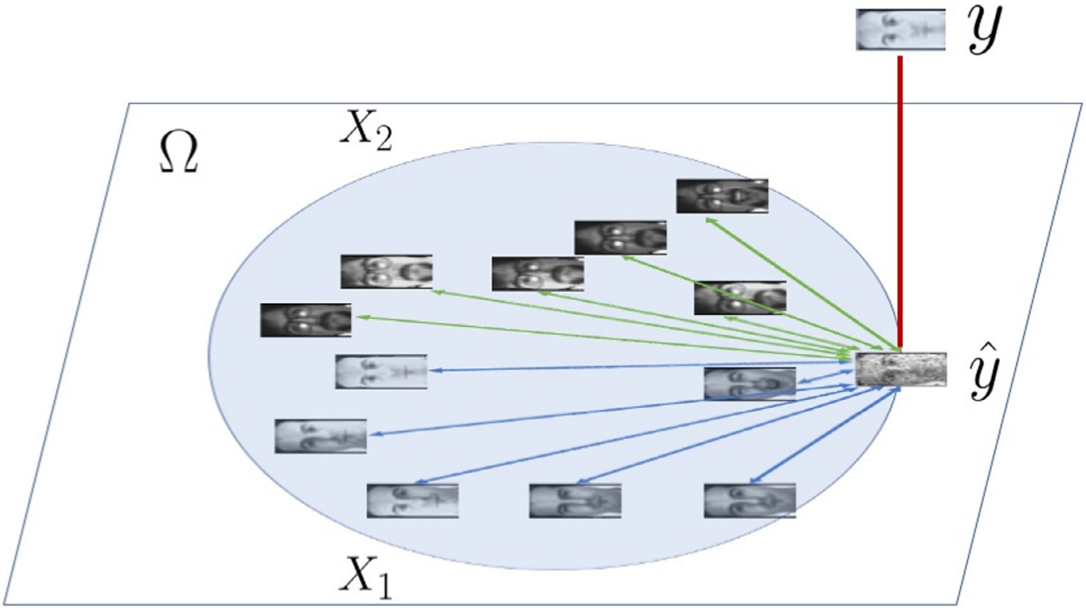

Fig. 1. Problem of shortage of training faces.

improve the performance of face recognition under noise. Other promising approaches for extracting features are descriptor-based algorithms such as LBPs [1], which extract both shape and texture data and store them in histograms of features. The main advantages of LBPs are that they are invariant under gray-scale changes and their computational costs are very low. Thus, Wolf [25] achieved a remarkable success in recognizing faces in unconstrained datasets. However, in practice the efficiency of LBPs deteriorates significantly because of random noise in the areas surrounding the face. Tan [23] presented LTPs which not only inherit the advantages of LBPs but also significantly reduce noise sensitivity. Tan's paper showed that LTPs outperform LBPs in dealing with difficult illumination conditions when tested on challenging databases.

3. Proposed approach

We propose HCRC, which was inspired by recent research on CRC [29]. The CRC-based classifier was proved to be a fast and robust non-parametric classifier, as mentioned above. However, we found some of the major drawbacks of this representation. In this section, we focus on resolving the problem posed by the limited number of available training faces. Thus, we propose HCRC to overcome this drawback related to an inadequate number of training samples and high dimensional features.

First, we denote the set of K classes of identities by $X = [X_1, X_2, X_3, \dots, X_K]$ where X_i is the subset of the i^{th} class. The number of columns in the data matrix X_i is the same as the number of training vectors of the i^{th} class. We also need a label set L_X of images in the data matrix X in the training processes. Our task is to find a new representation for an arbitrary facial feature vector y that is better than CRC so that it can be effectively represented by all the training vectors on the entire dataset as follows

$$y = X\alpha \tag{1}$$

where α is the representation vector. Ideally, a solution for the sparse vector α can be found by solving the l_2-norm minimization problem. Unfortunately, the algorithm to solve this optimization equation either fails because of an NP-hard problem or because it converges very slowly. Therefore, in realistic applications, we replace this difficult problem with a cost-effective approximation, such as CRC. The strategy of CRC is to find a close approximation $\hat{y}$ that falls into the face subspace Ω spanned by the training faces and can be linearly represented by these training faces. In other words, vector $\hat{y}$ is a projection of vector y, which normally falls inside the subspace Ω. In most cases, CRC gains a high face recognition accuracy rate because the testing face is presented by over-complete training subspaces and its projection falls completely in its corresponding subspace. However, we want to explore the manner in which CRC deals with the problem of a lack of training faces, which probably occurs in many biometric systems using face recognition because of the diversity of training samples. The testing face image y usually belongs to a high-dimensional facial space, which requires to cover a very large number of diverse training faces. Hence, vector y may easily fall out of the collaborative subspace Ω and its projection vector $\hat{y}$ may locate near the boundary of this collaborative subspace. In such a case, we evaluate the Euclidean distances from the projection vector $\hat{y}$ to the training vectors. A typical example for this situation is given in Fig. 1. In this example, we assume that

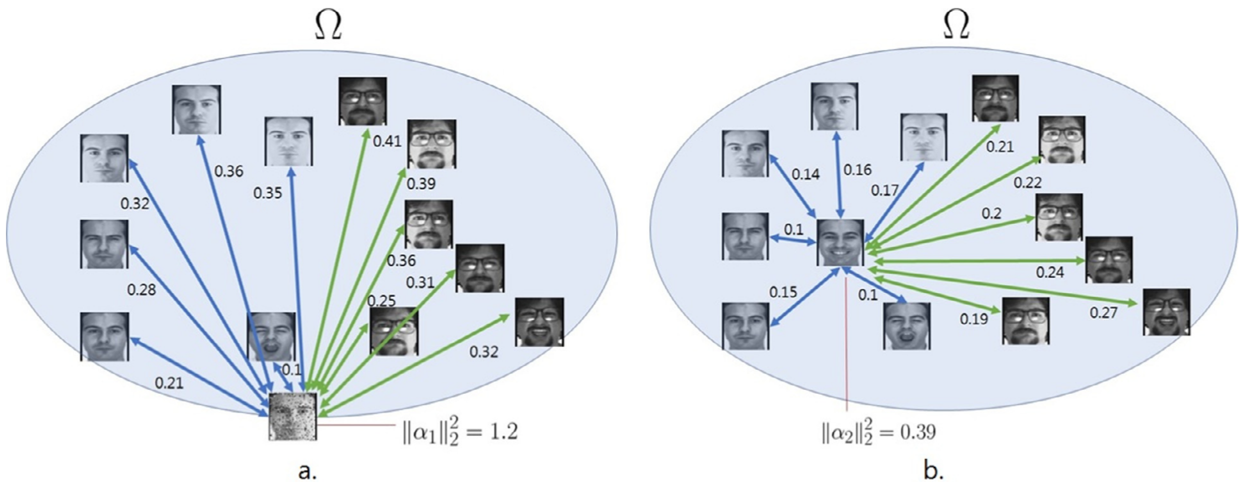

Fig. 2. Comparison of two typical positions of the projection vector $\hat{y}$. (a) The projection vector $\hat{y}$ falls near the boundary of the collaborative subspace Ω. (b) The projection vector $\hat{y}$ is located near the center of this subspace.

the testing face belongs to class X_1. Because of the lack of training faces to represent completely the testing face, the projection of the testing face $\hat{y}$ falls near the boundary of the collaborative subspace Ω instead of falling inside it. This shortage of training faces causes the poor representation of vector $\hat{y}$. Hence, the majority of training faces of class X_1 and class X_2 are far from vector $\hat{y}$. In this case, CRC fails to predict the identity of this testing face. The reason for this failure is that CRC focuses only on minimizing the Euclidean distance from the testing face y to its projection vector $\hat{y}$ on the collaborative subspace Ω while the Euclidean distances from this projection vector to training vectors are not taken into account. We explain more detail about this example by comparing two typical positions of the projection vector $\hat{y}$. One falls near the boundary of the collaborative subspace Ω, and the other is located near the center of this subspace, as shown in Fig. 2. The former position is presented by the vector $\alpha_1 = [0.21, 0.28, 0.32, 0.36, 0.35, 0.41, 0.39, 0.36, 0.31, 0.25, 0.32]$ with the l_2-norm square $\|\alpha_1\|_2^2 = 1.2$. The latter position is presented by the vector $\alpha_2 = [0.15, 0.1, 0.14, 0.16, 0.17, 0.21, 0.22, 0.2, 0.24, 0.27, 0.19]$ with the l_2-norm square $\|\alpha_2\|_2^2 = 0.39$. The results show that the l_2-norm square $\|\alpha_1\|_2^2$ of the representation vector α_1 is much bigger than the l_2-norm square $\|\alpha_2\|_2^2$ of the representation vector α_2. In addition, the coefficients of the vector α_2 are much smaller than those of the vector α_1. Thus, the representation vector α_2 is more reliable than the representation vector α_1 in identifying human faces. Moreover, it proves that if we can minimize the l_2-norm square of the representation vector, the projection of the testing face $\hat{y}$ falls closer to the center of the collaborative subspace Ω. Thus, the testing face can be recognized more precisely.

For dealing with this problem, an ideal solution is to provide a full training dataset for classification. However, it is difficult to build such a full training dataset because the collaborative subspace is too large to be covered. Thus, we propose to find an improved collaborative representation model that not only minimizes the Euclidean distance between the testing face y and the projection vector $\hat{y}$ but also minimizes Euclidean distances from this projection vector to the training classes. In a better collaborative representation model, we expect that the projection vector $\hat{y}$ moves as close as possible to the center of the collaborative subspace Ω and class *i*. For this reason, we propose a framework of learning a hierarchical collaborative representation-based classifier. We present this classifier in detail in the following section.

3.1. Hierarchical collaborative representation-based classification

In theory, the solution for the vector α can be found by using the extended formulation of the l_1-norm minimization problem

$$\begin{aligned} min\|\alpha\|_1 \quad \text{s.t.} \quad & \|y - X\alpha\|_2^2 \le \varepsilon_1 \\ & w_i\|X\alpha - X_i\alpha_i\|_2^2 \le \varepsilon_2 \quad i = 1, \ldots, K \end{aligned} \tag{2}$$

where w_i with $i = 1, \ldots, K$ represents regularization weights and ε_1 and ε_2 are small constants. The solution for this equation is a vector $\hat{\alpha}$, which is sufficiently sparse to classify the testing vector y in the collaborative subspace Ω. This optimization problem is subject to two types of constraint: the main constraint $\|y - X\alpha\|_2^2 \le \varepsilon_1$ and the additional constraints $w_i\|X\alpha - X_i\alpha_i\|_2^2 \le \varepsilon_2$ with $i = 1, \ldots, K$. As in the optimization problem of sparse representation as presented in [29], the main constraint $\|y - X\alpha\|_2 \le \varepsilon_1$ is aimed to optimize the coding vector α with the efficient approximate range ε_1 to account for the dense small noise in y. However, unlike in the sparse representation, we propose using the additional constraints $w_i\|X\alpha - X_i\alpha_i\|_2^2 \le \varepsilon_2$ to minimize the Euclidean distances from the coding vector $X\alpha$ to the coding vectors of training faces in each class X_i in the collaborative subspace Ω. In total, the additional constraints $w_i\|X\alpha - X_i\alpha_i\|_2^2 \le \varepsilon_2$ with $i = 1, \ldots, K$

are approximated in the range of ε_2 with the regularization weights w_i, which are automatically selected, depending on the prior knowledge and experience of each specific problem. We discuss the selection of these parameters in the following.

Although the solution for Eq. (2) can be found by using the l_1-norm minimization algorithm, this algorithm converges too slowly. In addition, since the collaborative representation plays a main role in improving the face recognition performance, the l_1-norm minimization algorithm can be entirely replaced by a weaker but more robust algorithm, the l_2-norm minimization algorithm, which shows a much lower complexity and an accuracy rate that is almost the same as that of the former one. As a result, the sparse representation of y by X can be formulated as

$$\hat{\alpha} = argmin_{\alpha}\left\{\|y - X\alpha\|_2^2 + \tau\|\alpha\|_2^2 + \textstyle\sum_{i=1}^{K} w_i\|X\alpha - X_i\alpha_i\|_2^2\right\} \tag{3}$$

where τ is the regularization parameter. Obviously, the original collaborative representation is a special case of our proposed algorithm when $w_i = 0$ with $i = 1, \ldots, K$. In fact, the selection of a better set of regularization weights w_i is important to make the projection vector $\hat{y}$ fall near the center of the collaborative subspace Ω and close to the class to which the face belongs. Thus, the accuracy of CRC is improved further. For this reason, we propose HCRC for our face recognition tasks. This classifier consists of two stages of recognition in which the regularization weights w_i are set to 0 in the first stage and updated in the next stage.

3.2. First stage classifier

Since the regularization weights w_i are set to 0 in the first stage, our first stage classifier is an original collaborative representation-based classifier. In fact, we can obtain two main advantages from the original collaborative representation-based classifier. On the one hand, the original collaborative representation-based classifier can be used to filter out quickly the majority of classes with a regularized residual higher than a threshold θ because the probability that the testing face y belongs to these classed is extremely low. On the other hand, this classifier also provides all the Euclidean distances from the projection vector $\hat{y}$ to the training classes, which are used to update the regularization weights w_i in the next stage.

In fact, the first stage classifier is a robust multi-class classifier to select a small number of the candidate classes to which the testing vector y most likely belongs. Then, we select a stronger classifier in the second stage to choose the best candidate from these classes. This strategy is aimed to reduce computational complexity and to achieve a higher recognition accuracy rate. The threshold θ is selected based on ratios of regularized residuals. In particular, we compute the regularized residuals of class i as

$$r_i = \|X \cdot \hat{\alpha} - X_i \cdot \hat{\alpha_i}\|_2^2 \tag{4}$$

where $\hat{\alpha_i}$ is the coefficient vector of class i. Class i and its training samples are excluded in the second stage if

$$\frac{r_i}{r_0} \geq \theta \tag{5}$$

is satisfied where r_0 is the minimal regularized residual.

Since the regularized residual of class i also shows how close the testing face y is to class i, we set the regularization weight w_i proportional to the regularized residual r_i with $i = 1, \ldots, K$. As a result, Eq. (3) can be modified as

$$\hat{\alpha} = argmin_{\alpha}\left\{\|y - X\alpha\|_2^2 + \tau\|\alpha\|_2^2 + \eta\textstyle\sum_{i=1}^{K} r_i\|X\alpha - X_i\alpha_i\|_2^2\right\} \tag{6}$$

During our experiments, we computed η based on the following equation to obtain the highest recognition accuracy

$$\eta = \frac{1}{r_0} \tag{7}$$

where r_0 is the minimal regularized residual. The regularization factor η is aimed to balance the parameters in Eq. (6) including τ and r_i with $i = 1, \ldots, K$. As a result, the first stage classifier collects a new set of K' classes of identities as $X' = [X'_1, X'_2, X'_3, \ldots, X'_{K'}]$ in which X'_i is the subset of the ith class. This small set of K' classes is classified by a stronger classifier in the second stage, and the set of the weight w_i, which is built in the first stage, is also used to improve this classifier's performance.

3.3. Second stage classifier

In this stage, we apply an improved collaborative representation method to encode the testing vector y over the training set X' by using the regularized least square method:

$$\hat{\psi} = argmin_{\psi}\left\{\|y - X'\psi\|_2^2 + \tau\|\psi\|_2^2 + \tfrac{1}{r_0}\textstyle\sum_{i=1}^{K'} r_i\left\|X'\psi - X'_i\psi_i\right\|_2^2\right\} \tag{8}$$

As in CRC, the solution for Eq. (8) is analytically derived:

$$\hat{\psi} = \left(X'^T X' + \tau \cdot I + \frac{1}{r_0}\sum_{i=1}^{K'} r_i(X'\psi - X'_i\psi_i)^T(X'\psi - X'_i\psi_i)\right)^{-1} X'^T y \tag{9}$$

Table 1
Architecture of our deep learning model.

Name	Type	Filter size/Stride	Output size
Conv1	Convolution	$5 \times 5/1$	$128 \times 128 \times 96$
Max1	Maxout	–	$128 \times 128 \times 48$
Pool1	Max Pooling	$5 \times 5/2$	$64 \times 64 \times 48$
Conv2	Convolution	$1 \times 1/1$	$64 \times 64 \times 96$
Max2	Maxout	–	$64 \times 64 \times 48$
Conv3	Convolution	$3 \times 3/1$	$64 \times 64 \times 192$
Max3	Maxout	–	$64 \times 64 \times 96$
Pool2	Max Pooling	$2 \times 2/2$	$32 \times 32 \times 96$
Conv3	Convolution	$1 \times 1/1$	$32 \times 32 \times 192$
Max3	Maxout	–	$32 \times 32 \times 96$
Conv4	Convolution	$3 \times 3/1$	$32 \times 32 \times 384$
Max4	Maxout	–	$32 \times 32 \times 192$
Pool3	Max Pooling	$2 \times 2/2$	$16 \times 16 \times 192$
Conv5	Convolution	$1 \times 1/1$	$16 \times 16 \times 384$
Max5	Maxout	–	$16 \times 16 \times 192$
Conv6	Convolution	$3 \times 3/1$	$16 \times 16 \times 256$
Max6	Maxout	–	$16 \times 16 \times 128$
Conv7	Convolution	$1 \times 1/1$	$16 \times 16 \times 256$
Max7	Maxout	–	$16 \times 16 \times 128$
Conv8	Convolution	$3 \times 3/1$	$16 \times 16 \times 256$
Max8	Maxout	–	$16 \times 16 \times 128$
Pool4	Max Pooling	$2 \times 2/2$	$8 \times 8 \times 128$
FC1	Fully Connection	–	512

In addition, we compute the regularized residuals of classes as follows

$$r'_i = \left\| X' \cdot \hat{\psi} - X'_i \cdot \hat{\psi}_i \right\|_2^2 \tag{10}$$

where $\hat{\psi}_i$ is the coefficient vector of class i. By finding the minimal regularized reconstruction error, the identity of y is computed as

$$\text{Identity}(y) = \arg\ \min_i \left\{ r'_i \right\} \tag{11}$$

4. Discriminative feature extraction

4.1. Fast deep feature learning

The purpose of a fast deep feature learning method is to build a deep model that transforms face images into a common set of distinct features. We aimed at building a deep convolutional network to train the best facial features. In our network, each training face image is resized to fit the input of the network, which is fixed at the size of 128 × 128 × 1. This proposed deep architecture is built based on eight fundamental convolution layers. Each convolution layer is connected to a maxout layer (Maxout), which is an improvement of the maxout network [7]. Unlike the maxout network, the maxout layer is considered the layer of maximal feature maps. In particular, each convolution layer is randomly categorized into n groups of feature maps. Feature values at the same coordinates from these groups are compared to select the maximal one, which is then assigned to the feature value at the same coordinates in the maxout layer. This maxout layer has some significant advantages in terms of improving face recognition performance. First, the maxout layer plays an important role in minimizing the number of necessary neurons and the number of network parameters in each layers. Second, a maxout layer is also considered a set of efficient activation functions, which are faster than traditional activation functions. These two advantages make this network much faster than other deep convolutional networks. Finally, but not least important, the maxout layer is used to quickly obtain a number of competitive features that are beneficial for building a good feature extraction model.

The architecture of our deep convolutional network is shown in Table 1. Four pooling layers using max filters are applied to down-sample feature maps and to reduce the number of learning parameters. This network includes a dropout layer, which is considered to constitute a good technique to prevent convolutional networks from overfitting. The ratio of dropping out connections in our network is set to 0.5. In the training stage, a softmax layer is added to generate an objective function. This deep model is then used as inputs to the HCRC-based classifier.

4.2. Very deep convolutional networks

Recently, the very deep convolutional network (VGG) [15] has been the state-of-the-art approach on ILSVRC classification and localization tasks [16]. It achieved an excellent face recognition performance on the challenging LFW database [9]. For this reason, we aimed to apply the VGG model for extracting the most discriminative features.

VGG exploits the effect of the convolutional network depth on the recognition accuracy rate. It is built based on a very deep architecture of convolution layers with very small convolution filters. By pushing the depth to weight layers, this model shows noteworthy improvement on face recognition accuracy. In the training process, the input to this model is an 224 × 224 RGB image that is passed through a stack of convolution layers containing only 3 × 3 convolution filters to capture the detail in the image. Some of these convolution layers are followed by five max-pooling layers that are used for reducing the number of parameters to learn and providing basic translation invariance. Three fully connected layers are added to the output of the stack of convolution layers so that all activations can be connected. Each of the first two fully connected layers contains 4096 channels, and the last fully connected layer outputs 1000 channels, corresponding with 1000 classes. The soft-max layer is the final output layer in this model, which is applied to perform multi-class classification such as face recognition. The configuration of this model was presented in [15].

4.3. Facial feature extraction using local ternary patterns

We focused in this study on effectively constructing high dimensional LTP descriptors that can store a large amount of discriminative features. This type of descriptor contributes to the improvement of the hierarchical collaborative representation-based classifier, because it can significantly reduce the effect of uncontrolled illumination, as well as being insensitive to random noise.

The LTP operator works in a 3 × 3 pixel block of a face image in which the difference between the center pixel and the neighboring pixel is encoded into a trinary code. We denote by l_c the gray level of the center pixel, and by l_p the gray level of the neighbors, where $p = 0, 1, \ldots, 7$. Thus, the LTP code is computed as

$$LTP = \sum_{p=0}^{7} f(l_p, l_c, th)3^p \tag{12}$$

Here, $f(l_p, l_c, th)$ is the threshold function

$$f(l_p, l_c, th) = \begin{cases} 1, \ l_p \geq l_c + th \\ 0, \ |l_p - l_c| < th \\ -1, \ l_p \leq l_c - th \end{cases} \tag{13}$$

where th is a threshold. If the threshold th is sufficiently large, a small gray change of the central pixel caused by noise can not change the codes for its neighborhood pixels in an image. This is the reason why an LTP is insensitive to noise in a face image. In our study, th was set to 5. In order to reduce the feature dimension, the LTP is constructed by an effective coding scheme. Each LTP is split into positive and negative LBP parts:

$$f_{po}(l_p, l_c, th) = \begin{cases} 1, \ l_p \geq l_c + th \\ 0, \ otherwise \end{cases} \tag{14}$$

$$f_{na}(l_p, l_c, th) = \begin{cases} 1, \ l_p \leq l_c - th \\ 0, \ otherwise \end{cases} \tag{15}$$

Thus, we applied the LTP operator to generate two positive and negative LBP images for extracting the face features.

In order to successfully conserve local features and to keep the spatial location information of the face, we construct a high dimensional feature of LTPs by dividing the face image into blocks. In our research, every block was fixed at a size of 8 × 8 pixels. The fundamental steps of our algorithm are shown in Fig. 3. The occurrences of LTP codes in each block are collected into a histogram. All these histograms are concatenated into a combined feature histogram, which consists of a large number of bins. In order to avoid the curse of dimensionality, we apply the method of principal component analysis (PCA) to transform this high dimensional histogram feature vector into a much lower dimensional feature vector. The output of PCA is a facial feature vector.

As in our deep learning model, outputs of the LTP model are used as inputs to the hierarchical collaborative representation-based classifier. This model is able to extract better features for real-time face recognition.

5. Experimental results and analysis

5.1. Dataset

In this section, we demonstrate the effectiveness of our proposed methods on the problems of face recognition, that are low-resolution faces, a limited number of training faces, noise, and occlusion. We used two challenging databases, the Extended Yale B [5] dataset and the AR [12] dataset, to evaluate the accuracy and the processing time of our methods, and compared our results with those of state-of-the-art algorithms. We also evaluated the recognition performance of our proposed methods and the state-of-the-art algorithms on the LFW-a database [9] in uncontrolled environments.

On these challenging databases, we also compared our proposed models with state-of-the-art deep models of learning facial features, including the CenterLoss network model (CenterLoss) [24] and the VGG network model (VGG) [15]. These

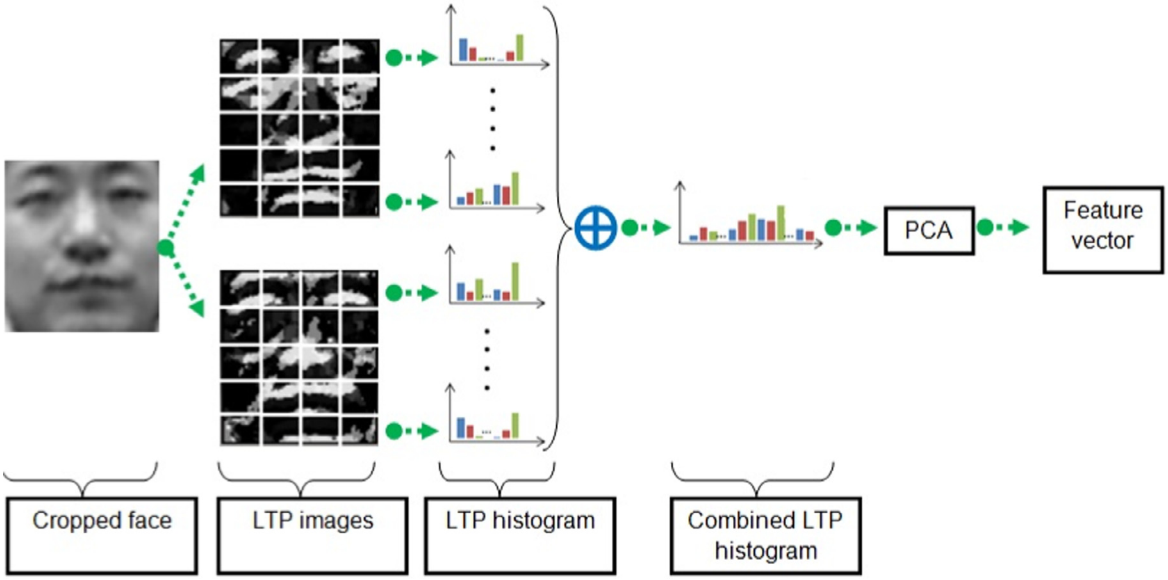

Fig. 3. Diagram of facial feature extraction.

Table 2
Comparison of proposed methods and other state-of-the-art methods on the AR database.

Method	Accuracy (%)	Method	Accuracy (%)
NN [2]	71.3	HCRC	95.7
LRC [14]	76.0	LTP-HCRC	**99.9**
SRC [26]	93.3	Maxout	85.5
CRC [29]	93.7	Maxout-HCRC	99.1
KCRC [11]	94.0	VGG [15]	98.6
MSPCRC [30]	96.4	VGG-HCRC	**99.9**
Volterra [10]	89.4	CenterLoss [24]	87.5
RCR [28]	95.9	CenterLoss-HCRC	99.7

are the best DCNN models for face recognition. Since models of these methods were published, we extracted deep features from these available models, compared them with our facial features, and used the results in our experiments. In all our experiments, our wide LTP model was built by collecting feature histograms of small blocks with a size of 12 × 12. For training our deep maxout network model, we collected training and testing faces from the MS-Celeb-1M dataset [8]. The learning rate of this model was set to 1e-3 initially and reduced to 5e-5 gradually. The weight decay of all convolutional layers is set to 0 and the ratio of the dropout layer to 0.5.

5.2. AR database

For evaluating the accuracy of face recognition in different environments, the AR database, which consists of 50 male and 50 female faces, was used for comparison purposes. For each subject, seven images which were different in illumination and expression, were collected for training, and another seven images were applied for testing. The images in this database were cropped and resized to 60 × 43 pixels. In the experiments on hierarchical collaborative representation-based classifiers, we set $\tau = \alpha = 1$. In the MSPCRC method, we used seven scales to obtain the best performance and the patch sizes were 10 × 10, 15 × 15, 20 × 20, 25 × 25, 30 × 30, 35 × 35, and 40 × 40. We conducted the first experiment to compare the recognition accuracy of our algorithms with that of the competing algorithms. In our experiments, Maxout, VGG and CenterLoss were used to extract deep facial features. We then applied HCRC for classifying these features. The performance of the proposed methods were then compared with the original deep networks, including our maxout network, the VGG network [15], and the CenterLoss network [24], respectively. The results of this comparison showed the contributions of the deep feature learning models and HCRC to improving the face recognition accuracy. We show a comparison of the algorithms in Table 2.

Table 2 shows that the classification accuracy of HCRC is better than that of the competing classifiers SRC and CRC. HCRC obtains a 2% performance gain as compared to CRC and is 2.4% more accurate than SRC. These results prove that HCRC effectively improves the performance of the original collaborative representation-based classifier. The results in Table 2 also in-

Table 3
Recognition rate for face images with random noise on the AR dataset.

Noise	10%	20%	30%	40%	50%
NN [2]	69.3%	63.7%	50.1%	31.7%	17.8%
LRC [14]	75.3%	70.9%	51.0%	36.4%	23.4%
SRC [26]	86.9%	75.7%	59.3%	42.9%	29.1%
CRC [29]	86.5%	74.6%	58.8%	42.0%	27.3%
KCRC [11]	87.2%	76.1%	59.4%	43.9%	28.0%
MSPCRC [30]	82.9%	70.4%	60.6%	47.9%	32.3%
Volterra [10]	85.0%	73.6%	56.7%	40.1%	24.9%
RCR [28]	91.5%	83.6%	68.9%	53.0%	35.6%
HCRC	90.1%	81%	66.8%	51.2	34%
LTP-HCRC	99.4%	98.2%	86.9%	49.5%	47.3%
Maxout	54.1%	24.6%	4.2%	2.4%	1.9%
Maxout-HCRC	75.2%	40.4%	14.8%	2.5%	2.1%
VGG [15]	65.5%	45.3%	14.0%	1.6%	1.1%
VGG-HCRC	82.0%	62.0%	27.3%	2.3%	1.9%
CenterLoss [24]	69.4%	48.0%	15.6%	3.7%	2.9%
CenterLoss-HCRC	86.8%	64.6%	31.8%	4.2%	3.5%

dicate that LTP-HCRC achieves an accuracy rate of 99.9%, and significantly outperforms other competitive approaches, which do not use deep learning models. These results indicate that the LTP model contributes very considerably to improving the recognition performance by removing noise from facial features. Furthermore, Table 5 shows that LTP-HCRC is not only highly accurate but also faster than its competitors and can be used in real-time applications of face recognition, such as surveillance security systems or mobile robots.

We also evaluated the performance of HCRC using deep facial features from the deep learning models, as shown in Table 2. Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC are more accurate than Maxout, VGG [15] and CenterLoss [24], respectively. This is because HCRC makes significant contribution to the accuracy improvement of classifying facial features in the deep learning models.

Table 2 also demonstrates that VGG-HCRC achieves 99.9% accuracy, which is the best among the competitive approaches using deep learning models, and is also as accurate as LTP-HCRC. Maxout-HCRC achieves a 99.1% accuracy rate, which is slightly less than that of VGG-HCRC and CenterLoss-HCRC. However, VGG-HCRC and CenterLoss-HCRC are approximately 7.8 times and 2.5 times, respectively, slower than Maxout-HCRC, because they have more network parameters. Maxout-HCRC would be much faster if it were run on more powerful devices with the support of GPUs. In general, Maxout-HCRC is a promising algorithm for real-time face recognition applications while VGG-HCRC is the best algorithm for face recognition systems that are not required to run in real time.

It should also be noted that the results of this experiment show that, although the performance of some deep neural networks such as VGG, and CenterLoss can even approach human performance on the challenging LFW dataset [9] in uncontrolled environments, LTP-HCRC still outperformed their recognition performance in this experiment. This can be explained by the problem of overfitting that we encountered when testing VGG, CenterLoss, and Maxout on our datasets. The problem of overfitting means that these models learned the detail and noise on a training dataset to the extent that it negatively affected the performance of these models on a new testing face. In particular, although these models performed very well on the training dataset by memorizing each training face, when they were faced with a new previously unseen testing face they did not have any general concepts on which to fall back.

In order to demonstrate further the key role of the feature extraction models, we conducted more challenging experiments using this dataset. We added more random noise to all the testing images. A number of pixels in each image was replaced by random values within [0, 255]. The percentages of pixels corrupted by random noise were set respectively to 10%, 20%, 30%, 40%, and 50% in the next experiment. For fair comparison, each experiment was run 20 times and we reported only the average results, which are shown in Table 3.

Table 3 demonstrates that HCRC is notably superior to MSPCRC and CRC under the different noise ratios. In particular, HCRC consistently outperforms MSPCRC, with an improvement of between 1.7% and 10.6%. The results in Table 3 also show that the LTP model still contributes to improving the recognition performance, because it is highly resistant to noise. Thus, LTP-HCRC achieves an accuracy of 99.4%, and is still by far the best approach. It even outperforms the other competitive approaches using deep learning models.

Unlike the results in the previous experiment, the accuracies of Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC drop dramatically when random noise increases in testing faces. Similarly to the case of the feature deep learning models in the previous experiment, this problem is also caused by overfitting. DCNNs are powerful for classification tasks, but they are not completely resistant to overfitting. In this case, the training dataset does not cover challenging face images, such as blurred, noisy and low resolution images. Consequently, the model can perform well on the training data but does not perform well on some evaluation datasets containing face images that it has never seen before.

Table 4
Recognition rate for face images with random occlusion on the AR dataset.

Occlusion ratio	10%	20%	30%	40%	50%
NN [2]	68.8%	65.2%	52.6%	29.5%	15.6%
LRC [14]	78.1%	75.6%	54.1%	40.0%	27.6%
SRC [26]	85.1%	73.9%	58.6%	40.2%	30.4%
CRC [29]	84.8%	74.2%	58.4%	39.1%	29.2%
KCRC [11]	86.3%	75.7%	60.5%	40.3%	30.7%
MSPCRC [30]	89.4%	79.9%	71.5%	63.9%	27.5%
Volterra [10]	83.4%	71.4%	53.8%	38.1%	25.6%
RCR [28]	88.5%	80.6%	68.3%	50.8%	34.3%
HCRC	88.8%	79%	68.5%	51.8%	40.1%
LTP-HCRC	99.6%	99%	97.1%	94.1%	86.1%
Maxout	76.1%	70.9%	56.5%	41.5%	17.6%
Maxout-HCRC	98.0%	95.3%	91.8%	82.9%	65.1%
VGG [15]	99.0%	97.6%	94.7%	80.7%	55.6%
VGG-HCRC	99.6%	99.0%	97.1%	86.4%	69.4%
CenterLoss [24]	81.4%	74.9%	59.9%	40.8%	23.6%
CenterLoss-HCRC	93.8%	87.8%	78.11%	60.9%	41.5%

Table 5
Recognition rate and processing time on the AR database in the first experiment.

Method	Recognition rate	Time
LTP-HCRC	**0.999**	**0.0455 s**
Maxout-HCRC	0.991	0.0972 s
VGG-HCRC	**0.999**	0.7509 s
CenterLoss-HCRC	0.997	0.2412 s

We also evaluated our proposed classifiers and their competing classifiers for face recognition with block occlusion, which makes the face recognition problem more challenging. In these experiments, each test image was randomly occluded by a small square block. Table 4 lists the experimental results of the challenging methods. It shows that HCRC outperforms CRC, being at least 4.0% more accurate. Table 4 also shows that LTP-HCRC achieves a high accuracy of recognition, and its accuracy is slightly reduced when the rate of occlusion dramatically rises in the testing face. LTP-HCRC is shown to be relatively insensitive to the corruption caused by occlusion. This proves that LTPs make a large contribution to the improvement of the recognition rate because of its robustness to different types of corruption, such as occlusion, illumination, noise, and shading.

Table 4 also compares our results of the approaches using DCNN models. It can be seen that Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC are effective for face recognition under occlusion, and are more accurate than Maxout, VGG, and CenterLoss, respectively. This can be explained by the fact that the testing face still keeps good features for recognition despite partial occlusion. Thus, although the rate of occlusion dramatically increases, the accuracies of Maxout-HCRC, VGG-HCRC and CenterLoss-HCRC are still high. Among these, VGG-HCRC is the best approach, and Maxout-HCRC achieves results comparable to those of CenterLoss-HCRC.

5.3. Extended Yale B database

We also used the Extended Yale B face database to evaluate the accuracy of our methods and their competitors under a wide variety of different illumination conditions. Changes in illumination conditions exerted one of the most significant effects on our face recognition results. The database consists of 38 identities under 64 illumination conditions. The face images were cropped and resized to 32×32 pixels. The recognition rates of our methods and competing methods on this database are shown in Table 6.

The results prove that LTP-HCRC and VGG-HCRC are the best of the algorithms included in this experiment. LTP-HCRC inherits the advantages of both HCRC, which shows a 0.9% performance gain over CRC, and LTPs, which are robust to noise and complex illumination conditions. Table 6 demonstrates that Maxout, VGG, and CenterLoss achieve high recognition accuracies. This is because these models can extract key facial features from this dataset. This is also the reason why VGG-HCRC achieves the highest recognition accuracy, while Maxout and CenterLoss are only 0.4% less accurate than VGG-HCRC.

As in the experiments we conducted using the AR face database, we once again corrupted testing images in the database to make the face recognition problem more challenging. Two types of corruption, random noise and random occlusion, were taken into account and the experimental settings were the same as in the previous experiments using the AR database.

The face recognition results for the dataset corrupted by random noise are listed in Table 7. As seen in this table, HCRC is consistently superior to CRC. The results also show that LTP-HCRC is the best of the algorithms included in this experiment.

Table 6
Comparison of proposed methods and other state-of-the-art methods on the Extended Yale B dataset.

Method	Accuracy (%)	Method	Accuracy (%)
NN [2]	91.6	HCRC	98.6
LRC [14]	95.9	LTP-HCRC	**99.5**
SRC [26]	97.9	Maxout	93.5
CRC [29]	97.9	Maxout-HCRC	99.1
KCRC [11]	98.1	VGG [15]	97.7
MSPCRC [30]	99.1	VGG-HCRC	**99.5**
Volterra [10]	93.7	CenterLoss [24]	93.5
RCR [28]	99.1	CenterLoss-HCRC	99.1

Table 7
Recognition rate for face images with random noise on the Extended Yale B dataset.

Noise	10%	20%	30%	40%	50%
CRC	84.9%	73.6%	53.3%	38.6%	25.0
HCRC	91.3%	83.5%	70.3%	53.7%	37.9%
LTP-HCRC	99.1%	98.9%	97%	84%	53.1%
Maxout-HCRC	84.0%	56.6%	24.3%	2.9%	2.1%
VGG-HCRC	89.9%	60.5%	28.3%	1.5%	1.3%
CenterLoss-HCRC	85.6%	57.4%	27.8%	9.2%	4.2%

Table 8
Recognition rate for face images with random occlusion on the Extended Yale B dataset.

Occlusion ratio	10%	20%	30%	40%	50%
LTP-HCRC	99.1%	98.7%	97.7%	91.5%	81.8%
Maxout-HCRC	99.1%	98.9%	96.8%	90.9%	75.1%
VGG-HCRC	99.3%	99.1%	98.6%	93.1%	79.2%
CenterLoss-HCRC	99.1%	99.0%	97.4%	92.9%	76.8%

In addition, we can also see that the accuracy of this algorithm is only slightly reduced while all others are significantly degraded. On the other hand, the accuracies of Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC reduce quickly when random noise increases in the testing faces. Once again, these results prove that the deep feature learning models are sensitive to random noise, and LTP-HCRC is still the state-of-the-art noise-resistant approach.

We then compared our proposed algorithms against competing algorithms in the Extended Yale B face database with block occlusion. The results are shown in Table 8. Although the corruption caused by occlusion is challenging, LTP-HCRC, Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC still achieve high recognition rates.

5.4. LFW-a database

The LFW-a database is the most challenging that we used to compare our algorithms with the competing methods. This database was built for studying unconstrained face recognition. It consists of 158 different individuals of different races, ages and genders. For each of these individuals, we collected five training and two testing images. Our goal was to evaluate the face recognition performance in unconstrained environments using a different number of training faces. All the faces in these images were cropped to 32 × 32 pixels, and those from the same individual differed in pose, expression, and illumination.

The numbers of training faces collected in the LFW-a database were set respectively to 1, 2, 3, 4, and 5 in the next experiments. We show recognition rate results in Table 9. As shown in the table, HCRC is more accurate than MSPCRC and CRC when the number of training faces is increased. Maxout-HCRC, VGG-HCRC, and CenterLoss-HCRC are more accurate than Maxout, VGG and CenterLoss, respectively. Among these, VGG-HCRC is the best approach, and Maxout-HCRC achieves results comparable to those of CenterLoss-HCRC. This can be explained by the fact that we can extract more complex feature sets by using a deep feature learning model than we might have using other machine learning tools. Furthermore, they inherit the advantages of the method of hierarchical collaborative representation-based classification, which outperforms the other representation-based methods. LTP-HCRC is slightly less accurate than Maxout-HCRC, VGG-HCRC and CenterLoss-HCRC.

To evaluate the noise-resistant property of the algorithms, we once again corrupted all the testing images on the database by adding more random noise. A number of pixels in each image was replaced by random values within [0, 255]. The percentages of pixels corrupted by random noise were set respectively to 10%, 20%, 30%, 40%, and 50%, as shown in Fig. 4. For each individual, we collected five training and two testing images. The recognition accuracy results are shown in Table 10. These results demonstrate that HCRC is notably superior to MSPCRC and CRC under the different noise ratios. The results

Table 9
Comparison of proposed method and other state-of-the-art methods on the LFW-a dataset.

Number of images	1	2	3	4	5
NN [2]	8.9%	13.7%	19.5%	20.6%	28.7%
LRC [14]	12.8%	23.9%	25.6%	29.3%	43.1%
SRC [26]	13.7%	24.7%	25.9%	30.7%	44.5%
CRC [29]	13.6%	23.4%	25.3%	30.4%	43.7%
KCRC [11]	13.9%	30.2%	37.5%	40.9%	44.1%
MSPCRC [30]	14.6%	35.0%	41.1%	46.0%	49.0%
Volterra [10]	12.9%	19.6%	21.4%	29.5%	41.56%
RCR [28]	16.4%	37.1%	45.4%	50.2%	53.7%
HCRC	14.6%	28.8%	35.3%	45.3%	51.0%
LTP-HCRC	20.6%	39.9%	49.7%	60.1%	67.1%
Maxout	17.5%	20.5%	23.5%	27.7%	31.5%
Maxout-HCRC	37.4%	45.7%	57.1%	67.3%	76.5%
CenterLoss [24]	21.0%	26.5%	27.7%	33.1%	35.4%
CenterLoss-HCRC	42.6%	50.3%	58.9%	69.4%	78.2%
VGG [15]	65.3%	75.7%	79.7%	82.3%	85.3%
VGG-HCRC	80.3%	88.2%	90.9%	94.1%	95.6%

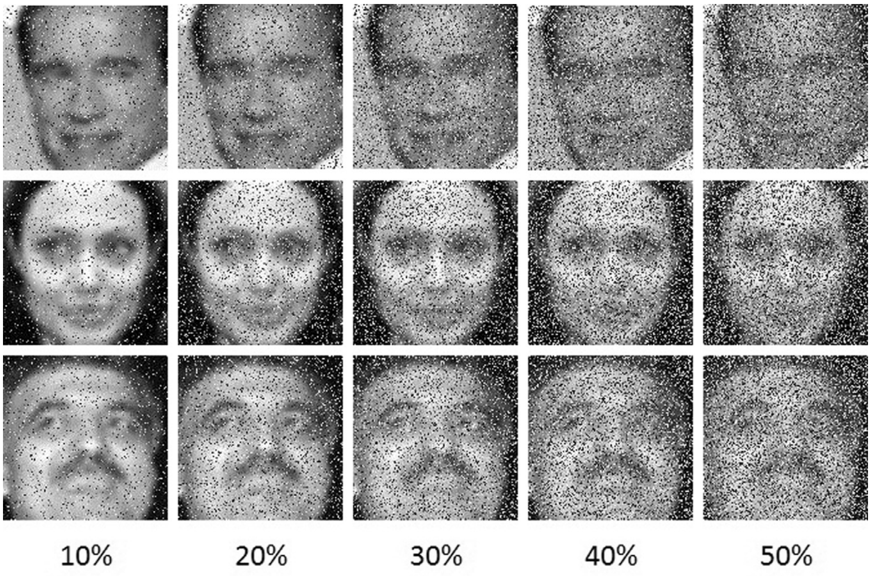

Fig. 4. Examples of testing images with random noise on the LFW-a dataset.

Table 10
Recognition rate for face images with random noise on the LFW-a dataset.

Noise	10%	20%	30%	40%	50%
NN [2]	15.8%	8.6%	2.4%	1.6%	1.0%
LRC [14]	27.6%	13.8%	4.6%	3.3%	1.1%
SRC [26]	31.3%	16.9%	6.9%	5.7%	2.4%
CRC [29]	30.0%	15.0%	5.4%	4.4%	1.9%
KCRC [11]	33.8%	17.8%	11.6%	8.3%	4.1%
MSPCRC [30]	39.7%	20.4%	12.3%	8.6%	5.2%
Volterra [10]	18.8%	9.7%	4.6%	2.7%	1.6%
RCR [28]	42.9%	24.5%	16.1%	10.9%	5.7%
HCRC	38.0%	16.8%	9.2%	7.6%	4.4%
LTP-HCRC	67.0%	53.8%	33.5%	17.4%	7.9%
Maxout	17.5%	5.5%	2.5%	1.0%	1.0%
Maxout-HCRC	22.5%	7.6%	3.5%	1.7%	1.6%
CenterLoss [24]	16.1%	3.5%	1.9%	1.1%	1.0%
CenterLoss-HCRC	18.0%	6.6%	4.4%	2.5%	1.6%
VGG [15]	30.1%	5.6%	1.1%	0.5%	0.5%
VGG-HCRC	35.8%	7.0%	1.7%	0.6%	0.6%

Table 11
Recognition rate for face images with random occlusion on the LFW-a dataset.

Occlusion ratio	10%	20%	30%	40%	50%
NN [2]	25.9%	20.1%	14.4%	6.7%	2.1%
LRC [14]	47.5%	43.3%	38.7%	28.7%	19.6%
SRC [26]	54.3%	50.7%	43.9%	35.1%	26.4%
CRC [29]	55.4%	50.3%	42.1%	32.3%	25.3%
KCRC [11]	57.6%	55.8%	49.9%	46.7%	40.4%
MSPCRC [30]	58.7%	58.3%	51.2%	49.0%	43.3%
Volterra [10]	52.7%	48.8%	39.8%	30.5%	21.7%
RCR [28]	60.5%	59.7%	50.3%	43.6%	53.7%
HCRC	59.7%	59.1%	47.8%	40.8%	31.0%
LTP-HCRC	67.1%	67.0%	66.8%	64.9%	54.1%
Maxout	51.8%	41.9%	25.5%	10.1%	3.9%
Maxout-HCRC	65.6%	52.2%	38.0%	18.4%	6.3%
CenterLoss [24]	34.9%	20.9%	13.7%	8.1%	3.5%
CenterLoss-HCRC	48.7%	35.4%	21.8%	15.5%	6.0%
VGG [15]	84.0%	79.6%	71.9%	57.8%	20.5%
VGG-HCRC	94.6%	88.9%	82.0%	66.5%	34.2%

in Table 10 indicate that LTP-HCRC achieves the highest accuracy and significantly outperforms the other competitive approaches using deep learning models. Once again, these results prove that the model of local ternary patterns is highly resistant to noise.

In contrast to LTP-HCRC, the accuracies of Maxout-HCRC, VGG-HCRC and CenterLoss-HCRC fall sharply when random noise increases in the testing face. Similarly to the results in the previous experiment, this is because of the overfitting problem that happens when the DCNN models deal with low-resolution testing images with high levels of noise. The results in this experiment show that the state-of-the-art deep feature learning models can not address effectively the face recognition challenges, such as very noisy, low-resolution images.

We then evaluated our proposed face recognition methods with occlusion. In this experiment, each testing image was randomly occluded by a square block with different sizes. Table 11 lists the experimental results of the competitive methods. These results show that HCRC outperforms CRC, being at least 4.3% more accurate. Table 11 also shows that VGG-HCRC achieves a remarkable recognition accuracy rate. The accuracy rate of VGG-HCRC decreases slowly when the rate of occlusion rises sharply in the testing face. LTP-HCRC is the second best method in this experiment. Maxout-HCRC yields lower accuracy rates than LTP-HCRC, and achieves results comparable to those of CenterLoss-HCRC.

6. Conclusion

In this paper, we proposed a hierarchical collaborative representation-based classifier for face recognition. This classifier consists of two stages of recognition in which the regularization weights are initialized in the first stage and updated in the next stage to improve the recognition rate. In addition, our classifier can be improved by combining it with the feature extraction models. There are two types of model we have adopted in this paper. First, we presented the DCNN models that was trained on a very large, challenging dataset of training faces to extract a number of distinctive features for face recognition. We built a Maxout network for real-time face recognition applications. In addition, we combined the state-of-the-art DCNN models with HCRC to improve the recognition accuracy rate. Second, we introduced a wide LTP model, which was proved to be insensitive to random noise and significantly reduce the effect of uncontrolled illumination.

HCRC was shown to significantly outperform other representation-based classifiers, because it is better to encode an arbitrary testing face over the whole training set by using additional constraints of the Euclidean distances from the approximator of the testing face to the training faces in each class.

For dealing with a wide variety of noise and illumination, our face recognition algorithm based on the combination of LTPs and HCRC is one of the most accurate algorithms. The experimental results on challenging datasets demonstrate that this algorithm significantly outperforms its competitors. This algorithm is not only accurate but is also sufficiently fast to be run in real time.

Coupling a hierarchical collaborative representation-based classifier with a DCNN model of discriminative feature learning is advantageous, because it addresses some challenges in unconstrained face recognition, such as pose variations and illumination changes. VGG-HCRC is the best among competitive approaches using deep learning models. Although our model, Maxout-HCRC, is slightly less accurate than VGG-HCRC, it is much faster than VGG-HCRC. Maxout-HCRC is a promising algorithm for real-time face recognition applications. The experimental results also show that all the three deep feature learning models, VGG-HCRC, Maxout-HCRC, and CenterLoss-HCRC, can not address the face recognition challenges, such as very noisy, low-resolution images.

In the future development of our methods, we intend to improve the performance of the hierarchical collaborative representation-based classifier by using l_1-norm, which has been proven to be more accurate than l_2-norm in dealing with partial corruption or occlusion, as mentioned in [26]. This is because the l_1-minimization solution is naturally sparser and can be applied to recover the sparsest solution when this solution is sufficiently sparse.

We also aim to develop a noise-resistant network for face recognition.

Acknowledgements

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (NRF-2015R1D1A1A01060422) and also by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information and Communications Technology Promotion (2015-0-00932).

References

[1] T. Ahonen, A. Hadid, M. Pietikäinen, Face recognition with local binary patterns, in: Computer Vision - ECCV 2004, Vol. 3021 of Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2004, pp. 469–481.
[2] N.S. Altman, An introduction to kernel and nearest-neighbor nonparametric regression, The American Statistician, 1992.
[3] S. Chopra, R. Hadsell, Y. LeCun, Learning a similarity metric discriminatively with application to face verification, in: Proc. CVPR, 2005.
[4] S.H. Gao, I.-H. Tsang, L.-T. Chia, Kernel sparse representation for image classification and face recognition, ECCV, 2010.
[5] A.S. Georghiades, P.N. Belhumeur, D.J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[6] C.I. Gonzalez, J.R. Castro, O. Mendoza, P. Melin, General type-2 fuzzy edge detector applied on face recognition system using neural networks, in: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2016, pp. 2325–2330.
[7] I.J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, Y. Bengio, Maxout networks, 2013. ArXiv:1302.4389.
[8] Y. Guo, L. Zhang, Y. Hu, X. He, J. Gao, Ms-celeb-1m: A dataset and benchmark for large-scale face recognition, CoRR, 2016. Abs/1607.08221.
[9] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Technical Report 07–49, University of Massachusetts, Amherst, 2007.
[10] R. Kumar, A. Banerjee, B.C. Vemuri, Volterrafaces: Discriminant analysis using volterra kernels, in: IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 150–155.
[11] W. Liu, L. Lu, H. Li, W. Wang, Y. Zou, A novel kernel collaborative representation approach for image classification, in: 2014 IEEE International Conference on Image Processing (ICIP), 2014, pp. 4241–4245.
[12] A.M. Martinez, The AR face database, 1998. CVC Technical Report, 24.
[13] P. Melin, C.I. Gonzalez, J.R. Castro, O. Mendoza, O. Castillo, Edge-detection method for image processing based on generalized type-2 fuzzy logic, IEEE Trans. Fuzzy Syst. 22 (6) (2014) 1515–1525.
[14] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE PAMI 32 (11) (2010) 2106–2112.
[15] O.M. Parkhi, A. Vedaldi, A. Zisserman, Deep face recognition, BMVC, 2015.
[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, CoRR, 2014. Abs/1409.0575.
[17] D. Sánchez, P. Melin, O. Castillo, Optimization of modular granular neural networks using a hierarchical genetic algorithm based on the database complexity applied to human recognition, J. Inf. Sci 309 (2015) 73–101.
[18] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[19] I.K. Shlizerman, S. Seitz, D. Miller, E. Brossard, The megaface benchmark: 1 million faces for recognition at scale, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[20] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Proceedings of Advances in Neural Information Processing Systems 27, 2014, pp. 1988–1996.
[21] Y. Sun, X. Wang, X. Tang, Deep learning face representation from predicting 10,000 classes, in: Proc. CVPR, 2014.
[22] Y. Sun, X. Wang, X. Tang, Deeply learned face representations are sparse, selective, and robust, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2892–2900.
[23] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Trans. Image Process. 19 (6) (2010) 1635–1650.
[24] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, ECCV, Springer, 2016.
[25] L. Wolf, T. Hassner, Y. Taigman, Descriptor based methods in the wild, Faces in Real-Life Images Workshop in ECCV, 2008, 2008.
[26] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[27] M. Yang, L. Zhang, J. Yang, D. Zhang, Robust sparse coding for face recognition, in: CVPR, 2011.
[28] M. Yang, L. Zhang, D. Zhang, S. Wang, Relaxed collaborative representation for pattern classification, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2224–2231.
[29] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: Which helps face recognition? in: Proceedings of the 2011 International Conference on Computer Vision, ICCV '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 471–478.
[30] P. Zhu, L. Zhang, Q. Hu, S.C.K. Shiu, Multi-scale patch based collaborative representation for face recognition with margin distribution optimization, in: Proceedings of the 12th European conference on Computer Vision - Volume Part I (ECCV'12), Springer-Verlag, Berlin, Heidelberg.