Multimed Tools Appl (2017) 76:13271–13299
DOI 10.1007/s11042-016-3708-4

Learning spherical hashing based binary codes for face recognition

Lei Tian[1] · **Chunxiao Fan**[1] · **Yue Ming**[1]

Received: 16 December 2015 / Revised: 10 May 2016 / Accepted: 24 June 2016 /
Published online: 12 July 2016

Abstract Local feature descriptor has been widely used in computer vision field due to their excellent discriminative power and strong robustness. However, the forms of such local descriptors are predefined in the hand-crafted way, which requires strong domain knowledge to design them. In this paper, we propose a simple and efficient Spherical Hashing based Binary Codes (SHBC) feature learning method to learn a discriminative and robust binary face descriptor in the data-driven way. Firstly, we extract patch-wise pixel difference vectors (PDVs) by computing the difference between center patch and its neighboring patches. Then, inspired by the fact that hypersphere provide much stronger power in defining a tighter closed region in the original data space than hyperplane, we learn a hypersphere-based hashing function to map these PDVs into low-dimensional binary codes by an efficient iterative optimization process, which achieves both balanced bits partitioning of data points and independence between hashing functions. In order to better capture the semantic information of the dataset, our SHBC also can be used with supervised data embedding method, such as Canonical Correlation Analysis (CCA), namely Supervised-SHBC (S-SHBC). Lastly, we cluster and pool these learned binary codes into a histogram-based feature that describes the co-occurrence of binary codes. And we consider the histogram-based feature as our final feature representation for each face image. We investigate the performance of our SHBC and S-SHBC on FERET, CAS-PEAL-R1, LFW and PaSC databases. Extensive experimental results demonstrate that our SHBC descriptor outperforms other state-of-the-art face descriptors.

✉ Yue Ming
myname35875235@126.com

Lei Tian
tianlei189@sina.com

Chunxiao Fan
fcxg100@163.com

[1] Beijing Key Laboratory of Work Safety Intelligent Monitoring, School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing, China

Keywords Face recognition · Local descriptor · Binary codes · Spherical hashing · Supervised spherical hashing

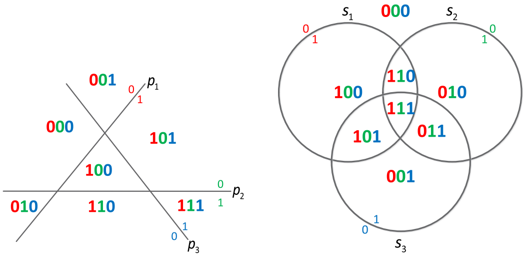
1 Introduction

During the past decades, face recognition has been successfully applied in many fields, it still attracts much attention due to its theoretical and practical challenges. Like those traditional pattern recognition problem, there are two critical problems on face recognition system: **feature representation** and **classifier training**. Recently, most of the existing works are focusing on these two aspects to improve the performance of face recognition methods when they faced with a variety of intra-class variabilities. For face representation, the purpose is to extract discriminative (global/local) features to make face images of different individuals more separable. For classifier training, the goal is to design an efficient (supervised/unsupervised) classifier to distinguish different face patterns.

Feature representation significantly affects the performance of face recognition system due to the large intra-class variations which reduce the similarity of face images from the same individual. Up to now, a various of face representation methods have been proposed, including holistic features [5, 6, 42] and local features [1, 23]. Though, these features have achieved great success for some controlled scenarios through designing low-level features elaborately, they can't achieve excellent performance when they faced with extreme intra-class variability and uncontrolled scenarios. Therefore, it is a challenging problem in face recognition that how to extract robust and discriminative feature when the intra-class variabilities are large.

Learning features from data itself instead of manually designing features is considered as a plausible way to overcome the limitation of hand-crafted features. Inspired by the research that binary codes are robust to local variations, we intend to encode high-dimensional original data into low-dimensional binary codes. In generally, the intra-variations is smaller than inter-variations of a particular individual in face recognition task. Therefore, the **motivation** of our proposed method is two-folds. On the one hand, our method is expected to assign the same binary codes for those data points which have shorter Euclidean distance on the original PDVs space. On the other hand, it can be expected to minimize the quantization error between binary codes and original PDVs so that the energy of real-valued PDVs is preserved as much as possible. In other words, our method is expected to assign the same binary code for the same individual, so that the intra-variations can be further reduced and the inter-variations can be enlarged.

In this paper, we propose a simple and efficient Spherical Hashing based Binary Codes (SHBC) to learn a discriminative and robust binary face descriptor in the data-driven way. Firstly, we extract patch-wise pixel difference vectors (PDVs) by computing the difference between center patch and its neighboring patches and grouping them as pixel difference matrix (PDM). Then, in order to reduce the quantization error between original PDVs and binary codes, and assign the same binary codes for those data points with shorter Euclidean distance, we perform spherical hashing based iterative optimization method to learn optimized binary face representations. Since hypersphere provides much stronger power in defining a tighter closed region than hyperplane, we perform the iterative binary learning method on hyperspheres instead of hyperplane. For each bit, hyperplane p_i and hypersphere s_i partitioning space into two parts (0 and 1), respectively. The partitioning examples are shown in Fig. 1. We can see that hypersphere-based hashing scheme encodes a higher number of closed regions than hyperplane-based one. This example can be generalized to higher



(a) hyperplanes-based hashing (b) hypersphere-based hashing

Fig. 1 Hypersphere-based hashing scheme encodes a higher number of closed regions than hyperplane-based one. (i.e., for 3-bits binary codes in the 2-D space, hypersphere-based hashing scheme encodes all $2^3 = 8$ binary codes, while hyperplane-based hashing scheme just encodes 7 binary codes). This figure is best viewed in color version

dimensional space which can demonstrate that hypersphere-based hashing scheme provides much stronger power in defining a closed region than hyperplane-based hashing scheme. Lastly, we cluster these binary vectors by traditional K-Means method to learn the dictionary and then pool them to obtain the histogram-based feature representations of input images. In addition, our SHBC also can be used with supervised Canonical Correlation Analysis (CCA), namely Supervised-SHBC (S-SHBC). Figure 2 illustrates the whole procedure of our SHBC feature learning method. Extensive experiments demonstrate our SHBC descriptors and S-SHBC descriptors outperform the most of state-of-the-art face representation methods.

Contributions The contributions of our work are summarized as follows:

1. We introduce the spherical hashing scheme into face recognition tasks. As above mentioned, the spherical hashing scheme provides much stronger power in defining a tighter closed region than hyperplane. Therefore, it is more likely to assign the same binary codes to the same individual under the condition that the energy of original real-valued PDVs is preserved as much as possible. So the intra-variations can be reduced and the inter-variations can be enlarged.
2. In order to improve the semantic understanding ability of our proposed method, we also propose a supervised version SHBC (S-SHBC) by using orthogonal basis projection method CCA to learn identity-bearing face feature representations.
3. We apply SHBC and S-SHBC to learn face features from a local face region, so that position-specific information of face image can be encoded in the learned binary features.
4. We perform extensive experiments on four face datasets to demonstrate the efficiency of our SHBC face descriptor. The results show that our proposed method are superior to most of state-of-the-art methods in both constrained and unconstrained face recognition scenarios.

The rest of the paper is organized as follows. Section 2 briefly reviews some recent related work of face recognition from most-related aspects. Then, we detail our proposed unsupervised SHBC and supervised SHBC feature learning method in the Section 3 and 4, respectively. We investigate the performance of our SHBC and S-SHBC method on face benchmark databases in Section 5 and conclude this paper in Section 6.

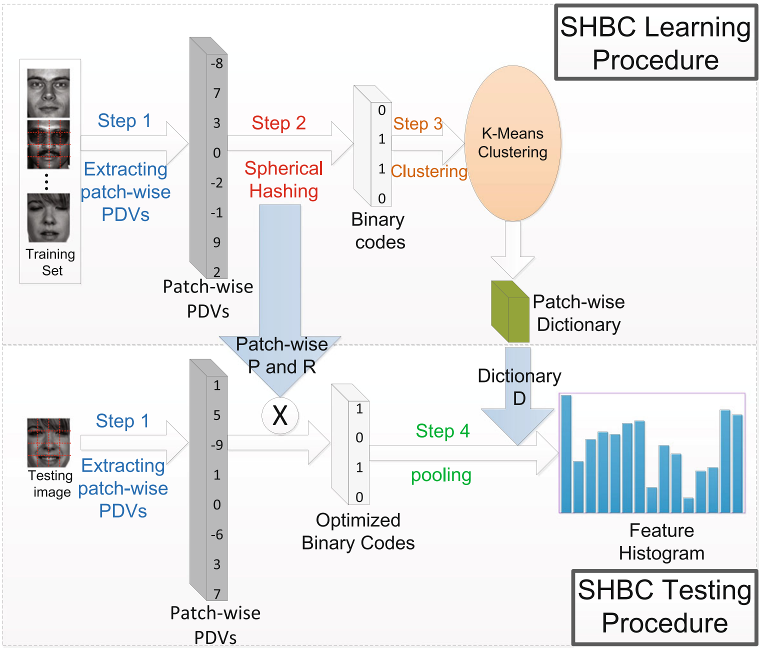


Fig. 2 The whole procedure of our proposed SHBC face representation method. For the learning stage, in the first step, we extract PDVs from each local region for each training face image. In the second step, we learn a discriminative feature mapping using unsupervised spherical hashing scheme, therefore, the extracted PDVs are projected into low-dimensional binary codes. In the third step, these learned binary codes are clustered to learn a dictionary by K-Means method. For the testing stage, we perform the same operation as the first step of learning stage. Then, the PDVs are encoded into binary codes using learned pivot matrix P and radius matrix R. Lastly, these binary codes are pooled as patch-wise histograms with the learned dictionary D and these local patch-wise histograms are concatenated to form the final representation of the test face image

2 Related work

Since the topics covered in face recognition literature are numerous, we focus on three most-related aspects about our work in this paper: **feature representation**, **feature learning** and **binary coding**.

2.1 Feature representation

During the past decades, there are extensive works on feature representation. The researchers usually classify these representations into two categories: holistic methods and local methods. Holistic feature representation regards the whole face image as a high-dimensional vector, and aims to map the high-dimensional vector into a low-dimensional feature subspace. The representative examples include PCA [42] and LDA [6]. However, the holistic feature representation is sensitive to local variability, such as occlusion, expression and misalignment.

Compared to the holistic face representation, local representation methods describe the pattern of each local block and concatenate these feature representations of local patch as the final output feature. The representative examples include LBP [1] and Gabor wavelet

[23, 44]. The Gabor wavelet can obtain the local feature representation of face image by computing specific spatial scales and spatial orientations. The LBP descriptor can capture the local structure of face image by computing the difference between the central pixel and its neighbors. A lot of LBP-based variant methods have been proposed in recent years [8, 10, 46, 50]. The LDGP [8] encodes the relationships between higher order derivatives of the neighbor pixel to reduces the computational time. The work [10] construct LBP-like vector by computing the adjacent pixels with diverse distances from different directions. Werghi et al. [46] proposes Mesh-LBP for computing local binary-like-patterns on a triangular-mesh manifold, which can extend most of the 2D-LBP variants to the mesh manifold. The work [50] extracts block-wise improved Weber local descriptor and encodes this descriptor by binary codes way. The combination of Gabor and LBP is also proposed for further improving performance in recent years, such as HGGP [54], LGXP [49] and GV-LBP [21] are representative examples. In these LBP-Gabor methods, a bank of Gabor filters with different scales and orientations are applied and the local pattern is encoded by LBP method.

However, the above local feature representation methods are all defined in a hand-crafted way, and they are usually require strong prior knowledge to be designed. In other words, it is hard for us to design an optimal encoding method by a hand-crafted way. Therefore, these hand-crafted features cannot be simply adopted for new test conditions and practical applications.

2.2 Feature learning

Learning features from the data itself instead of elaborately designing is considered as a way to avoid the problems of hand-crafted features. The feature learning methods usually are classified into two categories: low-level feature learning and high-level feature learning. There are lots of high-level feature learning methods proposed in recent years. Representative examples include convolutional neural network [39, 40, 51], deep auto-encoders [18] and PCA-Network [9, 41].

A great number of low-level feature learning methods also have been proposed in recent years [17, 22, 26, 29]. The DT-LBP method [29] is proposed by constructing a decision tree for each local region to learn the most discriminative LBP-like features. Hussain et al. [17] proposed a learning-based local quantized pattern (LQP) to encode the local binary or ternary pattern by adopting vector quantization. Lei et al. [22] proposed a discriminant face descriptor (DFD) by learning discriminative images filters and optimal soft sampling strategy using LDA criterion. In CBFD [26], researchers learn a feature mapping to project pixel difference vectors (PDVs) into a compact binary vector which evenly distribute at each learned bin. Compared to the hand-crafted feature representation methods, the learning-based feature descriptors which are learned by the data-driven way usually show better performances. Since learning-based feature representation methods can exploit more data-dependent discriminative information, they can achieve better performance when faced with more complicated test conditions.

2.3 Binary code learning

Compared to real-valued descriptors, binary codes methods have several properties. First, the binary codes are short so that enable limited memory to store large amounts of images. Second, the binary features have faster computational speed because they just have two values. Third, the binary coding methods are robust to local variations. In other words, we expect that binary code method can map images from the same person to binary strings with low feature space distance in the same time can map images from different person to binary strings with high feature space distance.

There are a lot of works about binary code learning in the field of computer vision. Spectral Hashing [45] preserves the similarity of original features by allocating bits based on separable Laplacian eigen-functions for image retrieval. Gong et al. [14] introduced a procrustean approach that minimizes the quantization error by iteratively rotating PCA-projected or CCA-projected data. Yu et al. [53] imposed a circulant structure on the projection matrix to speed up projections for high-dimensional data. Xia et al. [47] incorporated a sparse regularizer in an objective function to solve the over-fitting and expensive computation problem. There have been a lot of binary code methods proposed for randomized data embedding [3, 19, 34]. And these methods simply learn binary mapping by appropriately thresholding the projected data after randomized embedding.

However, most existing binary code learning methods were proposed for approximate nearest neighbor (ANN) search task or information retrieval task in computer vision. In fact, they are also competent for face recognition tasks. To our best knowledge, there is few related work before except CBFD method [26]. Therefore, in this paper, we applied spherical hashing scheme to learn discriminative and robust binary feature representation for face recognition task.

3 Spherical hashing binary codes learning

The motivation of spherical hashing binary codes learning is to learn a discriminative binary descriptor so that the intra-class difference is reduced and the inter-class difference is enlarged. Therefore, in this section, we first present the SHBC descriptor learning method, and then we introduce how to use SHBC descriptor for face representation.

3.1 Unsupervised SHBC descriptor learning

3.1.1 The extraction of PDVs

As mentioned above, there are some problems with existing binary codes methods, such as it is hard for them to sample large size neighborhoods due to high computational cost and it is difficult to design an optimal encoding strategy by hand-crafted way. Therefore, we propose an iterative optimization based binary learning method to solve the above limitations. Different from the traditional LBP-like method, we first extract the PDVs from a relatively larger pixel patch. The reason why we use PDVs instead of raw pixel patches to learn feature representation is that the PDVs can efficiently measure how pixel values change and describe the implicit local patterns in local face patches. The effectiveness of PDVs has been proven in the previous research work [22, 26].

Given training set $A = [a_1, a_2, \cdots, a_n]$, we extract many patch-wise PDVs from an original face image and group them as PDM $X = [x_1, x_2, \cdots, x_N]$, where $x_i \in \mathbb{R}^d$ is the ith PDV and N is the number of PDVs. Figure 3 illustrates how we extract PDM from neighboring patch of original image. We can see that the dimensionality d of one PDV is equal to $[(2R + 1) \times (2R + 1) - 1]$ and R is the sampling radius.

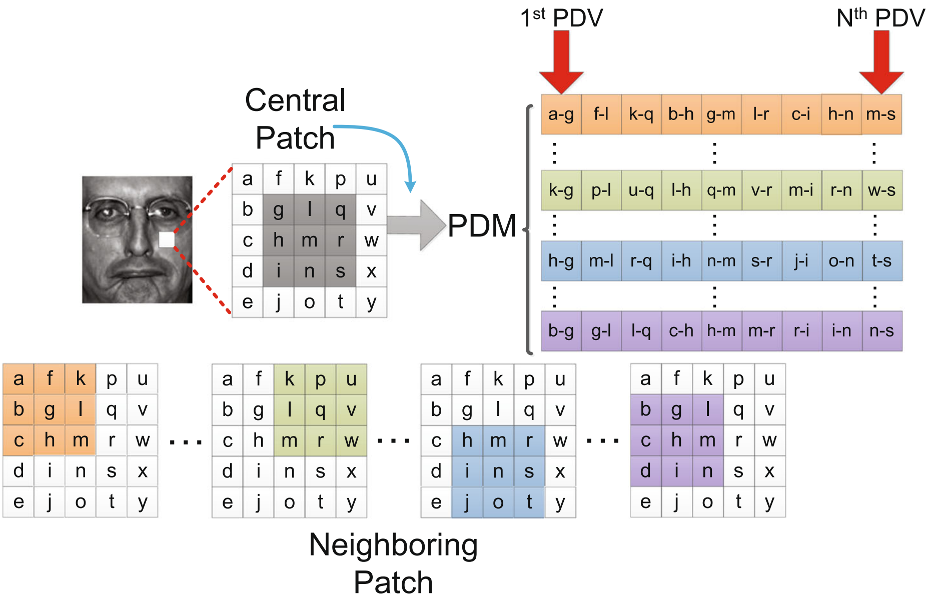

1st PDV
Nth PDV
Central Patch
PDM
a-g f-l k-q b-h g-m l-r c-i h-n m-s
k-g p-l u-q l-h q-m v-r m-i r-n w-s
h-g m-l r-q i-h n-m s-r j-i o-n t-s
b-g g-l l-q c-h h-m m-r r-i i-n n-s
a f k p u
b g l q v
c h m r w
d i n s x
e j o t y
Neighboring Patch

Fig. 3 The illustration of extracting a PDM from the face image. We sample the neighboring pixels in $(2R + 1) \times (2R + 1)$ space, where $R = 1$ is the sampling radius. Then the pixels of neighboring patches are compared with the pixel of central patch and the difference are used as the PDV. **In addition**, the arrangement order of the neighboring patches is insignificant, so we arrange the neighboring patches clockwise in this paper

3.1.2 Spherical hashing scheme

The purpose of our SHBC is to learn K spherical hashing functions to map and quantize each PDV x_i into $b_i = \left[b_{i,1}, b_{i,2}, \cdots, b_{i,K}\right] \in \{-1, 1\}^{1\times K}$. These functions are more likely to assign the same binary codes to those data points with shorter Euclidean distance (i.e., these data points probably come from the same individual) and minimize the quantization error between the learned binary codes and the real-valued PDVs. Therefore, for each spherical hashing function $h_i\,(x)$, we use a pivot $p_i \in \mathbb{R}^d$ and a hypersphere radius $r_i \in \mathbb{R}^+$ to define it:

$$h_i\,(x) = \begin{cases} -1 & when\ d\,(p_i, x) > r_i \\ +1 & when\ d\,(p_i, x) \le r_i \end{cases}, \tag{1}$$

where $d\,(p_i, x)$ denotes the Euclidean distance between pivot p_i and data point x. It is clear that the value of each spherical hashing function $h_i\,(x)$ indicates whether the data point x is inside the hypersphere.

In order to define a closed region in a d-dimensional space, at least $d + 1$ hyperplanes are need, and only one hypersphere is enough to form a closed region in arbitrarily high dimensional space. For the same number of hashing functions, hyperspheres can construct higher number of bounded regions than hyperplanes. In addition, the distances between data points located in the same region which are constructed by hypersphere are bounded. For face recognition task, **this property generates a merit that a region indexed by the binary code of the query data point can contain more promising candidates which from the same individual.** In order to verify the above conclusion, we also measure the maximum distance between any two data points that have the same binary codes, and take the average of the maximum distance among different codes on *dup2* subset of FERET dataset. The result is shown in Fig. 4. One can see that the bound regions of hypersphere are more tightly compared to hyperplane which are defined by Locality-Sensitive Hashing (LSH) [2] and learning based Iterative Quantization (ITQ) [12]. Compared to LSH and ITQ, spherical hashing functions are more likely to assign the same binary codes to two data points which have shorter distance on the original space. In addition, for hyperplanes based methods, the impact of learning scheme on the ability of bound region is negligible in the case of long codes. That is the reason why we use spherical hashing method to map PDVs instead of hyperplane-based methods.

3.1.3 The optimization goal of spherical hashing

As mentioned above, in order to make the learned binary codes discriminative and compact, we enforce two important criterions to learn these binary codes:

1. The balanced bits partitioning of data points can lead to the learned binary codes evenly distribute at each bin. Therefore, the histogram based features are more compact and informative [26].
2. Since independent hashing functions can assign points in a balanced way to different binary codes, the learned hashing functions should be independent between each pairs of them.

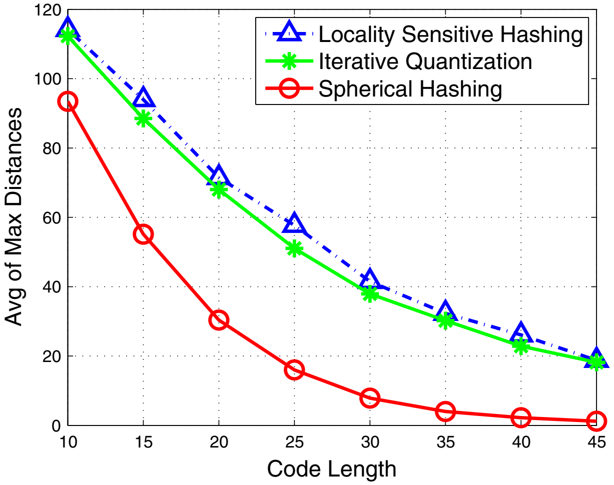


Fig. 4 This figure shows the average of the max. Euclidean distance among points having the same binary code with different code lengths based on hyperplane and hypersphere. For face recognition task, this value is directly related to the accuracy of method. We conduct this experiment on FERET database. Since ITQ and SHBC are generally only applicable for the case of $K \leq d$ and we are unaware of how to generalize the solution of these methods for the case of $K > d$, therefore, we can only show their ability of bound regions in the case $K \leq d$

For the first criterion, we define each hashing function $h_i(x)$ to have the equal probability for $+1$ and -1 as the following:

$$\Pr[h_i(x) = -1] = \Pr[h_i(x) = +1] = \frac{1}{2},\ x \in X,\ 1 \le i \le K. \tag{2}$$

For the second criterion, in order to clearly describe the property of independence between hashing functions, we define a probabilistic event V_i to represent the case of $h_i(x) = +1$. Therefore, two probabilistic events V_i and V_j are independent when $\Pr\left[V_i \cap V_j\right] = \Pr[V_i] \cdot \Pr\left[V_j\right]$. Once we achieve the first criterion (i.e., (2)), then the second criterion can be specified to the following equation:

$$\Pr\left[h_i(x) = +1, h_j(x) = +1\right] = \Pr[h_i(x) = +1] \cdot \Pr\left[h_j(x) = +1\right] = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \tag{3}$$

In fact, the (3) just represent the pair-wise hashing functions independence, and pair-wise independence does not guarantee the higher-order independence among more than two hashing functions. We also formulate the higher-order independence to satisfy these criterions shown in (2) and (3), however, we found that these higher-order independence does not provide the improvement of face recognition accuracy.

3.1.4 The iterative optimization process

To achieve the above criterions (the constraints shown in (2) and (3)), we propose an iterative process for computing K different hyperspheres. As the first phase of our iterative optimization process, we initialize pivot position of a hypersphere to be the median of randomly chosen multiple samples, shown as following:

$$p_i^0 = \frac{1}{g}\sum_{j=1}^{g} q_j,\ 1 \le i \le K, \tag{4}$$

where q_j are randomly selected points from the training set X, g is the number of such points and p_i^0 is the initial pivot of the ith hashing function. Of course, we also can initialize the pivots p_i^0 $(i = 1, \cdots, K)$ with randomly chosen K data points in the training set X, we found that the iteration process converges slightly quicker when the pivot p_i^0 are initialized by the median of randomly chosen multiple samples. And the experimental results show that the manner of initialized pivot slightly affect the results of face recognition task. The Table 1 shows the rank-one accuracy and average convergence time of each image on FERET dataset. In practice, we determined the parameter g as 5 using the cross-validation strategy on the FERET dataset.

Table 1 The rank-one accuracy and average convergence time of each face image comparison of different initialized pivots way

Initialization ways	*fb*	*fc*	*dup1*	*dup2*	Time (ms)[1]
Median	99.8%	100.0%	94.9%	93.6%	53.65
Random	99.8%	100.0%	93.5%	92.7%	57.12

[1] This experiments is conducted on a PC with 2.0 GHz i5 CPU and 64 GB RAM using matlab implementation

Then, as the second phase of optimization process, we refine the pivots and radii of hyperspheres by an iterative optimization process. To clearly describe the computational process, we define the following two variables v_i and $v_{i,j}$:

$$\begin{aligned} v_i &= \left|\left\{s_g | h_i\left(s_g\right) = +1, 1 \le g \le n\right\}\right|, \\ v_{i,j} &= \left|\left\{s_g | h_i\left(s_g\right) = +1, h_j\left(s_g\right) = +1 \le g \le n\right\}\right|, \end{aligned} \tag{5}$$

where $|\cdot|$ is the cardinality of the given set. The v_i measures how many data points in the training set X have $+1$ bit for ith hashing function, and it is used to satisfy the first criterion (2). The $v_{i,j}$ measures the number of data points pairs in the training set X whose hashing functions (i.e., $h_i(\cdot)$ and $h_j(\cdot)$) both are equal to $+1$. It is used to satisfy the second criterion (3). Therefore, our optimization goal is defined as the following:

$$\begin{aligned} v_i &= \Pr\left[h_i(x) = +1\right] \cdot n = \tfrac{n}{2}, \\ v_{i,j} &= \Pr\left[h_i(x) = +1, h_j(x) = +1\right] \cdot n = \tfrac{n}{4}. \end{aligned} \tag{6}$$

In order to meet our optimization goal, we use a two-stage iterative optimization method.

Fix radius $\mathbf{r_i}$ and update pivot $\mathbf{p_i}$ When radius r_i is fixed, we first adjust the pivot positions p_i^c and p_j^c of two hyperspheres in the cth iteration, so that $v_{i,j}$ becomes closer to $\frac{n}{4}$. For each pair of two hyperspheres i and j, when $v_{i,j}$ is greater than $\frac{n}{4}$ (i.e., the pivots of those two hyperspheres are too closer), a repulsive "force" is applied to placing them farther away. Otherwise, an attractive "force" is applied to pushing them closer. The repulsive or attractive "force" from p_j^c to p_i^c can be defined by the following

$$f_{j\to i} = \frac{1}{2}\frac{v_{i,j} - n/4}{n/4}\left(p_i^c - p_i^c\right). \tag{7}$$

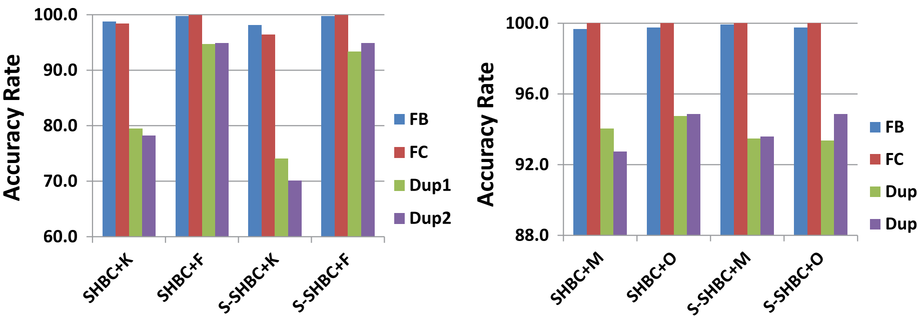
For the pivot p_i^c, the resulting force f_i is defined as the average of all forces which computed from other all pivots:

$$f_i = \frac{1}{K}\sum_{j=1}^{K} f_{j\to i}. \tag{8}$$

Then, the pivot p_i^c is updated as $p_i^c = p_i^c + f_i$.

Of course, we can also use K-Means to compute the pivots of hyperspheres. The comparison of force-based update pivots approach (i.e., SHBC+F and S-SHBC+F) and K-Means-based update pivots approach (i.e., SHBC+K and S-SHBC+K) on FERET dataset are shown in Fig. 5a. We can clearly see that K-Means is not suitable for updating the pivots of hyperspheres. Since this alternative does not ensure the independence between hashing functions, those pivots obtained by K-Means are close to the data mean in high dimensional space. Therefore, a large part of the regions are not covered by any hypersphere because the learned hyperspheres are highly overlapped.

Fix pivot $\mathbf{p_i}$ and update radius $\mathbf{r_i}$ Then, once pivots are computed, we can update the radius r_i of the ith hypersphere to make v_i become closer to $\frac{n}{2}$. Of course, we can simple set each radius r_i as $d\left(p_i^c, x_{n/2}\right)$, where training samples X are sorted in terms of distance from the pivot p_i^c and $d(\cdot,\cdot)$ denotes the Euclidean distance between two data points. However, when the single data point $s_{n/2}$ is located in a dense region, this simple radius-update approach could lead to undesirable partitioning. Therefore, we relax the first criterion so that the radius r_i can maximize the margin from data points to the surface of hypersphere. For our max-margin based radii optimization method, we first sort samples X



(a) Update Pivots Methods (b) Update Radii Methods

Fig. 5 The accuracy rate comparison of different update Pivots and Radii approach on FERET dataset. **a** **SHBC+F** denotes the force-based update pivots approach is used in SHBC, **SHBC+K** denotes the K-Means-based update pivots approach is used in SHBC. **b** **SHBC+O** denotes the max-margin based optimization update radii approach is used in SHBC, **SHBC+M** denotes the median-based update radii approach is used in SHBC

into $x_1, x_2, \cdots, x_n$ according to $d\left(p_i^c, x_j\right)$. Then, we select a set of candidate indices near the median $n/2$ for the max-margin based optimization:

$$J = \left\{ j \middle| \left(\frac{1}{2} - \alpha\right) n \leq j \leq \left(\frac{1}{2} + \alpha\right) n,\, j \in \mathbb{Z}^+ \right\}, \tag{9}$$

where α is a parameter that controls the degree of relaxation and we set $\alpha = 0.05$ in our all experiments. The optimized index $\hat{j}$ which maximizes the margin from data points to the hypersphere is defined as following:

$$\hat{j} = \underset{j \in J}{\arg\max} \left[d\left(p_i^c, x_{j+1}\right) - \left(p_i^c, x_j\right) \right] \tag{10}$$

Then, the radius of hypersphere r_i is defined as follow:

$$r_i = \frac{1}{2} \left[d\left(p_i^c, x_{\hat{j}}\right) + d\left(p_i^c, x_{\hat{j}+1}\right) \right]. \tag{11}$$

We compare the max-margin based optimization update radius approach (SHBC+O and S-SHBC+O) and median-based update radius approach (SHBC+M and S-SHBC+M) on FERET dataset, and the experimental results are shown in Fig. 5a. We can observe that max-margin based optimization update radius approach slightly improves the performance of SHBC and it hardly increase computational cost. Therefore, we apply it to update the radii of hyperspheres in our work.

Finally, we perform our iterative process until the learned hyperspheres do not make further improvements in terms of satisfying two criterions. We consider the mean and standard deviation of $v_{i,j}$ as a indicator of convergence of iterative process. Ideal values for the mean and standard deviation of $v_{i,j}$ are $n/4$ and 0, respectively. In order to avoid over-fitting, we stop our iterative process when the mean and standard deviation of $v_{i,j}$ are within error tolerances $\delta_m\%$ and $\delta_s\%$ of the ideal values of $v_{i,j}$. *Algorithm 1* summarizes the whole procedure of our proposed IQBC binary learning method.

Algorithm 1 Spherical Hashing Binary Codes Iterative Optimization Process

```
Input: The training set $A = [a_1, a_2, \cdots, a_n]$, the error tolerances $\delta_m\%$, $\delta_s\%$, the length of
    $K$ and the sampling radius $R$.
Output: Optimized pivot positions $p_1, p_2, \cdots, p_K$ and optimized radii $r_1, r_2, \cdots, r_K$ for
    $K$ hyperspheres.
Step 1 (Extraction of PDVs): Extracting the pixel difference vectors from $A$ with
    sampling radius $R$ and obtaining PDM $X = [x_1, x_2, \cdots, x_N]$;
Step 2 (Iterative Optimization):
    2.1. Initialization:
    Initialize $p_1, p_2, \cdots, p_K$ with $p_i^0 = \frac{1}{g}\sum_{j=1}^{g} q_j$ for quicker convergence
    Initialize radii $r_1, r_2, \cdots, r_K$ using max-margin radii update method (Eq.(11))
    Initialize $v_{i,j}$ for each pair of hashing functions $(h_i(\cdot), h_j(\cdot))$
    2.2. Iterative Process:
    repeat
        2.2.1. Fix radii and update pivot:
        for $i = 1$ to $K - 1$ do
            for $j = i + 1$ to $K$ do
                $f_{j \to i} = \frac{1}{2}\frac{v_{i,j} - n/4}{n/4}(p_i - p_j)$, $f_{i \to j} = -f_{j \to i}$
            end for
        end for
        for $i = 1$ to $K$ do
            $f_i = \frac{1}{K}\sum_{j=1}^{K} f_{j \to i}$
            $p_i = p_i + f_i$
        end for
        2.2.2. Fix pivot and update radii:
        Update $r_1, \cdots, r_K$ using max-margin radii update method (Eq.(11))
        2.2.3. Compute Convergence Indicator:
        Compute $v_{i,j}$ for each pair of hashing functions $(h_i(\cdot), h_j(\cdot))$
    until $avg\left(\left|v_{i,j} - \frac{n}{4}\right| \le \delta_m \frac{n}{4}\right)$ and $std(v_{i,j}) \le \delta_s \frac{n}{4}$
Step 3 (Output): Optimized pivot positions $p_1, p_2, \cdots, p_K$ and optimized radii
    $r_1, r_2, \cdots, r_K$ for $K$ hyperspheres.
```

3.2 Clustering and pooling

Having learned the optimized pivot positions $p_1, p_2, \cdots, p_K$ and optimized radii $r_1, r_2, \cdots, r_K$ for K hyperspheres by spherical hashing scheme, the PDVs can be projected into a low-dimensional binary feature. In order to make our learned binary codes more data-adaptive, we first apply an unsupervised clustering method to clustering learned binary features. We found that different clustering methods do not affect the experimental result, therefore, we choose the K-means to learn a dictionary from the training set for its simplicity and efficiency. At last, each projected binary code is pooled by using the learned dictionary, and the histogram-based features are used as the final face representation.

3.3 Unsupervised SHBC based face representation

Previous researches have shown that different face regions have different position-specific discriminative information [17, 22, 26]. In order to extract useful co-occurrence of pattern from face image, we propose to learn our SHBC descriptor from each local region. First, we divide each face image into several non-overlapped local regions. Then, we use the spherical hashing scheme to learn SHBC descriptor for each local region and concatenate local feature representation as a histogram-based output feature. At last, the simple Nearest Neighbor classifier (NN) can be adopted to measure the similarity of different face images. Another issue worth mentioning is that the dimensionality reduction method like WPCA can be used to further improving the performance of our descriptor.

4 Supervised spherical hashing binary codes learning

When label information is available for training samples, we can incorporate the label information into our SHBC model so that it can improve the semantic understanding of SHBC method, named as Supervised SHBC (S-SHBC). Inspired by the work [12], we use CCA as supervised method in our work. It is an effective method for extracting a common latent space from data view and label view.

We assume that the SHBC binary feature representation of the ith training image $f_i \in \{0, 1\}^K$ has associated with it a label vector $y_i \in \{0, 1\}^t$, where t is the total number of labels. A given entry of y_i is set to 1 if the face image is belong to the corresponding label and other elements are set to 0. Therefore, the binary descriptor f_i and label vector y_i can form the rows of feature matrix $F \in \{0, 1\}^{N\times K}$ and label matrix $Y \in \{0, 1\}^{N\times t}$. The goal that we incorporate CCA into SHBC is to find projection directions w_k and v_k for binary feature vector and label vector so that the correlation between the projection data Fw_k and Yv_k is maximized:

$$\begin{aligned} &C\left(w_k, v_k\right) = w_k^T F^T Y v_k \\ &s.t. \quad w_k^T F^T F w_k = 1, \quad v_k^T Y^T Y v_k = 1. \end{aligned} \tag{12}$$

According to [12], the projection of CCA can be obtained by solving the following generalized eigenvalue problem:

$$F^T Y \left(Y^T Y + \varepsilon I\right)^{-1} Y^T F w_k = \lambda_k^2 \left(F^T F + \varepsilon I\right) w_k, \tag{13}$$

where ε is a small regularization constant used to prevent the trivial solution and we set it to be 10^{-4} in our experiment. Once we have w_k, we also can solve the corresponding v_k. But in the task of face recognition, we assume that label information will be unavailable at test stage, we only care about the projection directions in the data space. Inspired by the finding that the most of discriminative information usually concentrate in the top eigenvectors for supervised embedding methods, we scale those trailing eigenvectors by the corresponding eigenvalues. Therefore, we can form a projection matrix W whose columns are given by the scaled eigenvectors w_k. Finally, we obtain the supervised binary feature $\hat{F} = FW$ which preserves both visual and semantic (label) similarity.

5 Experiments

We evaluate our SHBC and S-SHBC descriptors on the benchmark database, such as FERET, CAS-PEAL-R1, LFW and PaSC. The FERET [32] and CAS-PEAL-R1 [11] datasets are used to demonstrate the effectiveness of our proposed descriptor for face recognition in controlled scenarios. The LFW [16] dataset is employed to show the generalization ability (i.e., learn SHBC or S-SHBC model from controlled scenarios and test them on uncontrolled scenarios) of our proposed descriptor in uncontrolled scenarios. And the PaSC [7] dataset is employed to show the excellent performance of proposed descriptor in unconstrained scenarios.

5.1 Evaluation on FERET

The FERET dataset consists of 13,539 face images of 1,565 subjects. The complete dataset is partitioned into six disjoint sets: *training, gallery, fb, fc, dup1, dup2*. The *training* set contains 1,002 face images, and the *gallery* set consists of 1,196 images of 1,196 subjects and probe sets (*fb, fc, dup1, dup2*) include expression, illumination and aging variations, respectively. Here, all face images in these six subsets are aligned and cropped into 128×128 pixels according to the provided eye coordinates. Figure 6 shows some aligned and cropped example images from the FERET dataset.

We first perform SHBC (or S-SHBC) feature learning method on the *training* set, then we extract feature representation from the other five subsets by using learned hashing functions. In our experiments, we set the dictionary size to be 500 and the number of local region to be 8 × 8. Therefore, we extract a 32,000-dimensional (8 × 8 × 500) feature from each face images by using SHBC (or S-SHBC) descriptor. The iteration number is set to 50, which is determined by using cross-verification strategy on the *training* set. At last, the Whitening PCA (WPCA) which is conducted on the *gallery* set only is applied to reducing the dimension of SHBC (or S-SHBC) feature into 1,000 and NN classifier with the cosine distance is used for face matching.

5.1.1 Parameter analysis

In this section, we first investigate the impact of various parameters in our SHBC and S-SHBC method, and find suitable values of various parameters.

A. Impact of the binary codes length In this part, we first study the impact of different binary codes length K in our SHBC and S-SHBC methods on FERET dataset. We set sampling radius r to be 3 and a 48-dimensional patch-wise PDV is extracted from each pixel, the error tolerances δ_m and δ_s is empirically set as 0.10 and 0.15, respectively. We further study the impact of sampling radius r, error tolerances δ_m and δ_s in the following parts. We vary the binary codes length K from 10 to 40 with step size 5. Figure 7 shows the average recognition rate of SHBC and S-SHBC versus different binary codes length K. The results show that the value of binary codes length K have not a significant effect on the performance of face recognition, and SHBC achieves better performance than S-SHBC on larger codes length. In order to balance accuracy rate and training efficiency of our SHBC and S-SHBC, we set binary codes length K to be 15 in the following experiments.

Fig. 6 Aligned and cropped face examples from the FERET dataset

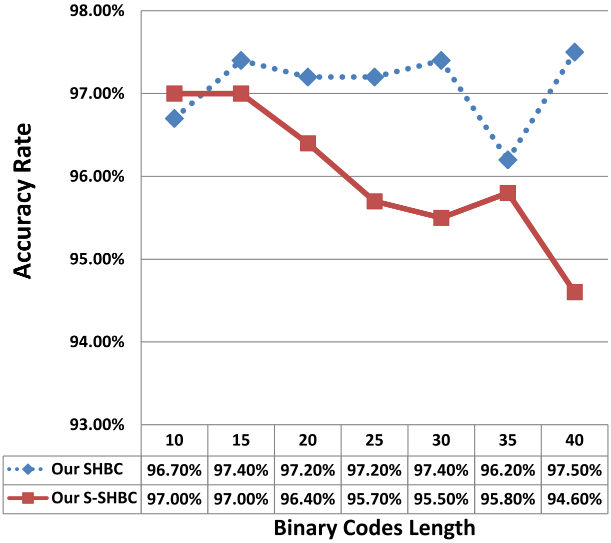
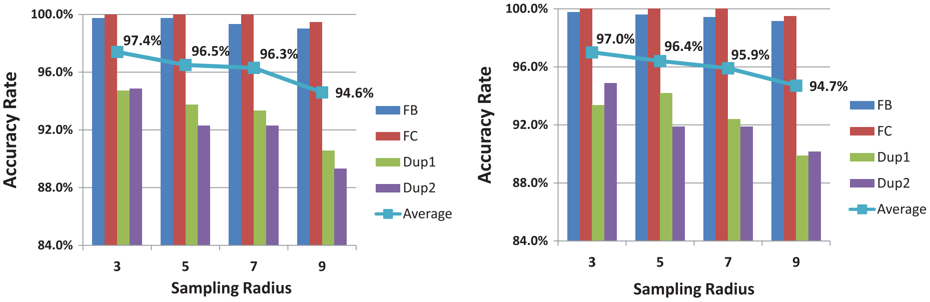


Fig. 7 The average rank-one recognition rate of SHBC and S-SHBC on different FERET subsets versus different binary codes length

B. Impact of the sampling radius We further study the impact of sampling radius r of SHBC and S-SHBC in this part. We set the binary codes length K to be 15 and the error tolerances δ_m and δ_s is empirically set as 0.10 and 0.15, respectively. We vary the sampling radius r from 3 to 9 with step size 2. Figure 8 shows the accuracy of SHBC and S-SHBC versus different sampling radius r. We can observe that our SHBC and S-SHBC method can achieve excellent performance when sampling radius r is set between 3 to 7, and the best recognition rate is obtained when r is 3. Therefore, we set the sampling radius r of both SHBC and S-SHBC methods to be 3 in subsequent experiments.

C. Impact of the error tolerances At last, we study the impact of error tolerances δ_m and δ_s of SHBC and S-SHBC in this part. We set the binary codes length K to be 15 and the sampling radius r to be 3. Five error tolerances pairs (δ_m, δ_s) are tested, and the



(a) SHBC (b) S-SHBC

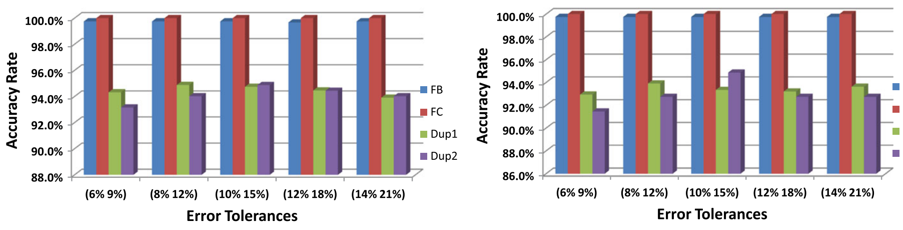
Fig. 8 The rank-one recognition rate of SHBC and S-SHBC versus different sampling radius r on FERET dataset

results are shown in Fig. 9. The results show that the values of error tolerances δ_m and δ_s have non-negligible affect on *dup1* and *dup2* subsets. We found that the smaller error tolerances pairs (δ_m, δ_s) can cause an over-fitting and larger error tolerances pairs (δ_m, δ_s) can cause an under-fitting. Therefore, in order to avoid undesired over-fitting or under-fitting, we choose an appropriate error tolerances pairs ($\delta_m = 0.10$, $\delta_s = 0.15$) in the following experiments.

5.1.2 Comparison with the state-of-the-art face recognition method

In this section, we investigate the performance of SHBC and S-SHBC method when it faced with various intra-class variation. We compare our SHBC descriptor with popular local face descriptors like LBP [1], LDP [55], DT-LBP [29], LQP [17], DFD [22], CBFD [26] and so on. The SHBC and S-SHBC descriptors are learned from the FERET training set, and all methods are performed on the *gallery* set and four standard testing subsets. We set the binary codes length K is set to be 15, the sampling radius r to be 3 and the error tolerances δ_m and δ_s to be 0.10 and 0.15, respectively. Table 2 lists the face recognition rate of our proposed SHBC (S-SHBC) and other state-of-the-art face descriptors on the FERET database. The experimental results demonstrate that

1. The performance of hand-crafted methods generally inferior to learning-based methods. This is because our SHBC and other learning-based methods can extract more discriminative and data-adaptive feature representations from raw data than those hand-crafted descriptors.
2. Compared with those recently proposed learning-based real-value coded feature descriptors, such as DT-LBP, LQP and DFD, the learning-based binary coded feature descriptors achieve higher recognition rate, such as our proposed SHBC and CBFD. Since binary codes have stronger robustness to local intra-class variabilities, the better performance are obtained by our SHBC method.
3. With WPCA, our proposed SHBC method achieves the **best** face recognition performance on **all** probe sets. Especially, our SHBC improves the performance of DFD by 2.9 and 2.6 percent on *dup1* and *dup2* probe set, and improves the performance of CBFD by 1.2 percent and 1.7 percent on *dup1* and *dup2* probe set. It demonstrates that SHBC can extract discriminative and robust binary feature representations from face images.
4. The performance of S-SHBC is slightly poor than SHBC, but it is still better than other learning-based methods like DFD, LQP and POEM. It achieves the **best** face recognition rate on *fb*, *fc* and *dup2* probe sets and the **third highest** recognition rate on *dup1*



(a) SHBC (b) S-SHBC

Fig. 9 The rank-one recognition rate of (**a**) SHBC and (**b**) S-SHBC versus different error tolerances pairs (δ_m, δ_s)

Table 2 Recognition Rate (%) Comparison with state-of-the-art face descriptors with the standard FERET Evaluation Protocol

Methods	*fb*	*fc*	*dup1*	*dup2*	Year
LBP [1]*	97.0	79.0	66.0	64.0	2006
LGBP [57]*	98.0	97.0	74.0	71.0	2005
LVP [30]*	97.0	70.0	66.0	50.0	2006
LGT [20]*	97.0	90.0	71.0	67.0	2007
HGGP [54]*	97.6	98.9	77.7	76.1	2007
HOG [31]*	90.0	74.0	54.0	46.6	2008
DT-LBP [29]*	99.0	100.0	84.0	80.0	2009
LDP [55]*	94.0	83.0	62.0	53.0	2010
GV-LBP-TOP [21]*	98.4	99.0	86.0	85.0	2011
DLBP [28]*	99.0	99.0	86.0	85.0	2011
I-LQP [17]*	99.2	69.6	65.8	48.3	2012
PEOM [43]*	97.0	95.0	77.6	76.2	2012
DFD [22]*	99.2	98.5	85.0	82.9	2014
CBFD [26]*	98.2	100.0	86.1	85.5	2015
SHBC	98.2	99.5	85.0	83.3	–
S-SHBC	98.0	99.0	83.7	80.8	–
LBP+WPCA [1]*	98.5	84.0	79.4	70.0	2006
I-LQP+WPCA [17]*	**99.8**	94.3	85.5	78.6	2012
POEM+WPCA [43]*	99.6	99.5	88.8	85.0	2012
DFD+WPCA [22]*	99.4	**100.0**	91.8	92.3	2014
CBFD+WPCA [26]*	**99.8**	**100.0**	93.5	93.2	2015
SHBC+WPCA	**99.8**	**100.0**	**94.7**	**94.9**	–
S-SHBC+WPCA	**99.8**	**100.0**	93.4	**94.9**	–

*Note that the results are from the original papers

The bold values denote the best performance on one subset or one dataset

probe set. These experimental results demonstrate that our S-SHBC method is also able to extract more discriminative feature than previous methods and is robust to various intra-class variations.

5. It is worth noted that there is **less than average 3 samples** for each subject on the *training* set of FERET dataset. The problem of **Small Samples Size** leads to S-SHBC method is not able to extract more discriminative feature than SHBC. However, the performance of S-SHBC will achieve as excellent as SHBC even outperform than SHBC when the larger scale training samples with label are provided (see the Section of 5.2).

5.1.3 Comparison with other binary codes learning methods

As mentioned above, most of existing binary codes learning methods were proposed for ANN search, though they are also competent for face recognition task. Therefore, in this section, we employ other four binary codes learning methods which are widely used for visual search to learn face feature representation. They are LSH method [2], Spectral Hashing (SH) [45], one-layer Anchor Graph Hashing (1-AGH) and two-layer Anchor Graph Hashing (2-AGH) [24], Angular Quantization-based Binary Coding (AQBC) [13] and ITQ method [12]. There are two comparable methods in [12]: PCA-RR and PCA-ITQ. PCA-RR quantizes the PCA-projected data into binary codes with a random orthogonal rotation matrix and PCA-ITQ quantizes the PCA-projected data with an optimized rotation matrix found by iterative rotation. We implement all these seven methods by using original codes provided by the authors. These seven binary codes learning methods are implemented in our model by just replacing the objective function of our SHBC (i.e., (2) and (3)) with the objective function of other binary codes methods, and other steps are consistent with *Algorithm 1* so that our SHBC and S-SHBC can be fairly compared with other binary codes learning methods. The parameters of different binary codes learning method are the same as those used in the above experiments. Table 3 lists the recognition rates of our methods and other binary codes learning methods on FERET.

We can observe that our SHBC and S-SHBC outperform other state-of-the-art binary codes learning methods in face recognition task. Since our SHBC-like method can provide strong power in defining a tighter closed region than hyperplane based methods (such as ITQ-like and LSH), so that our methods are more likely to assign the same binary codes to two data points which have shorter Euclidean distance on the original space. Therefore, the testing images can be correctly classified and the better performance is achieved by our SHBC-like methods.

5.1.4 Computational cost

At last, we compare the computational time of different LBP-like feature extraction methods, including original LBP [1], C-LBP [15], LGBP [57], DFD [22], CBFD [26]. In our implementation, all face images are divided into 8 × 8 = 64 non-overlapping blocks. All the experiments are computed on a PC with 2.0 GHz CPU and 64 GB RAM using matlab implementation. Table 4 shows the feature dimension and average extraction time of various methods. Though the feature dimension of LGBP is higher than learning-based methods (i.e., DFD, CBFD and our SHBC), its performance is worse than learning-based methods. It again demonstrates the discriminative power and efficiency of learning-based methods. It is clear that the computational efficiency of our SHBC is nearly same as CBFD, and higher than LGBP and DFD. Compared to DFD, our SHBC extracts only one PDV for each pixel,

Table 3 The Recognition Rates (%) of Our Proposed SHBC, S-SHBC and Other Binary Codes Learning Methods on FERET Dataset

Methods	*fb*	*fc*	*dup1*	*dup2*	avg.
LSH	99.4	**100.0**	93.4	90.6	95.6
SH	99.7	**100.0**	91.7	90.6	95.5
1-AGH	99.0	99.0	82.0	82.5	90.6
2-AGH	98.9	98.5	83.1	82.5	90.8
AQBC	99.7	99.1	93.5	92.3	96.2
PCA-RR	99.6	**100.0**	93.8	91.2	96.2
PCA-ITQ	99.6	**100.0**	93.9	94.4	97.0
SHBC	**99.8**	**100.0**	**94.7**	**94.9**	**97.4**
S-SHBC	**99.8**	**100.0**	93.4	**94.9**	97.0

The bold values denote the best performance on one subset or one dataset

Table 4 Computational Time (ms) Comparison of Different LBP-like Face Representation

Methods	Feature dimension	Extraction Time (ms)
LBP [1]	3,776	88.9
C-LBP [15]	11,328	95.1
LGBP [57]	655,360	9414.0
DFD [22]	50,176	2604.5
CBFD [26]	32,000	299.9
SHBC	32,000	305.9

and DFD extracts a set of PDVs for each pixel according to predefined neighborhood pattern. Therefore, the number of learned PDVs in our SHBC is less than DFD, so the feature extraction time of our SHBC is less than DFD.

5.2 Evaluation on CAS-PEAL-R1

There are 9,060 face images from 1,040 subjects in CAS-PEAL-R1 dataset, which provides large scale face images with various intra-class variabilities, including pose, expression, accessory and lighting. In this section, we evaluate the performance of our SHBC when it faced with these intra-class variabilities. In our experiments, we follow the standard evaluation protocols. We use five subsets of the CAS-PEAL-R1 in our experiment, including *training*, *gallery*, *expression*, *accessory* and *lighting*. The *training* set contains 1,200 images of 300 subjects, and the *gallery* set contains 1,040 face images from 1,040 subjects. We use *expression*, *accessory* and *lighting* set as the probe set. They contains 1,570, 2,285 and 2,243 images, respectively. All face images in these five subsets are aligned and cropped into 150 × 130 pixel according to the provided eye coordinates. Figure 10 shows some aligned and cropped examples from CAS-PEAL-R1 dataset.

We first use *training* set to learn our SHBC model, then perform feature extraction on *gallery* and other three probe sets. Finally, the WPCA is applied to reduce the feature dimension into 1,039, followed with the NN classifier with cosine metric. The parameters of our SHBC are the same as those used in the above experiment. Table 5 tabulates the recognition

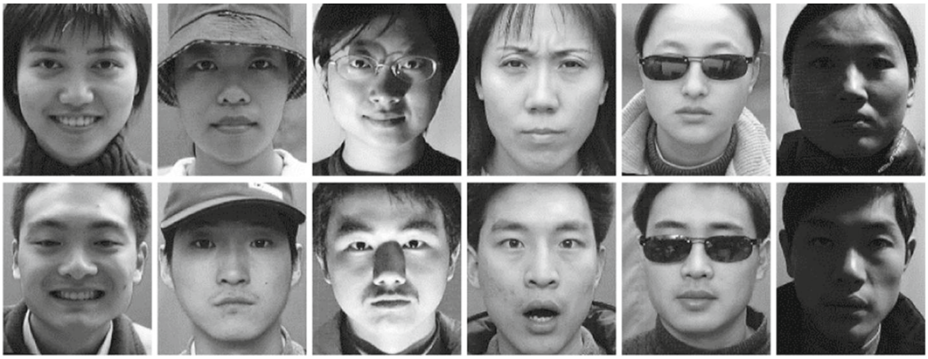


Fig. 10 Aligned and cropped face examples from the CAS-PEAL-R1 dataset

Table 5 Recognition Rate (%) Comparison with our SHBC and the State-of-the-art Face Descriptors on CAS-PEAL-R1

Methods	*Expression*	*Accessory*	*Lighting*	Year
LGBP [57]*	95.0	87.0	51.0	2005
LVP [30]*	96.0	86.0	29.0	2006
HGGP [54]*	96.0	92.0	62.0	2007
LLGP [48]*	96.0	90.0	52.0	2009
DT-LBP [29]*	98.0	92.0	41.0	2011
DLBP [28]*	99.0	92.0	41.0	2011
DFD [22]*	99.3	94.4	59.0	2014
CBFD [26]*	99.4	94.8	59.5	2015
SHBC	99.4	95.7	80.3	–
S-SHBC	99.2	95.2	75.4	–
DFD+WPCA [22]*	99.6	96.9	63.9	2014
CBFD+WPCA [26]*	99.7	97.2	67.4	2015
IFL+WPCA [25]*	99.3	96.5	64.3	2015
JFL+WPCA [25]*	99.7	97.2	67.4	2015
SHBC+WPCA	99.7	**97.7**	**81.1**	–
S-SHBC+WPCA	**99.8**	97.6	76.5	–

*Note that the results are from the original papers

The bold values denote the best performance on one subset or one dataset

rate of our methods and other feature descriptors on the CAS-PEAL-R1 dataset. We can get the following observations:

1. Compared with previously proposed methods, our proposed methods again achieve the highest recognition performance on all probe sets.
2. The SHBC with WPCA achieves the highest recognition rate on *accessory* and *lighting* probe sets. Especially, SHBC improves the best previous CBFD+WPCA and JFL+WPCA results by 13.7 percent on *lighting* set. To our best knowledge, **no single feature descriptor with a simple classifier can achieve such excellent performances on *lighting* set**. It demonstrates that our SHBC method is able to extract more discriminative and robust feature representation than previous methods, especially faced with *lighting* scenarios.
3. The S-SHBC method also achieves excellent performance on three probe sets when four training samples for each subject are provided, especially on *expression* set (mainly because the *training* set provides more expression variations than other variations). Since S-SHBC incorporates the class label information into SHBC model in the learning process, the incorporation of visual information and label information helps to extract more discriminative feature representation.
4. When the training samples are four for each subject, the performance of S-SHBC is close to SHBC, even outperform than SHBC on *expression* set. We can expect that the performance of S-SHBC can be further improved if larger scale (i.e., >4 samples/subject) training samples with label are provided.

5.3 Evaluation on LFW

Labeled Faces in the Wild (LFW) dataset are collected from the web for studying the unconstrained face recognition problem. There are 13,233 images from 5,749 different subjects, which consist of large pose, expression, illumination, aging and occlusion variations. There are 3,000 matched face pairs and 3,000 mismatched pairs which are equally divided into 10 folds on the *view 2* dataset. There are six evaluation setting on this dataset [16]. We consider the *unsupervised setting* evaluation to be the best choice for evaluating the discriminative ability of learned features, because it does not depend on metric learning and classifier model training.

In order to evaluate the discriminative ability and generalization ability (i.e., learned from constrained images and test on unconstrained images) of our methods when faced with real unconstrained scenarios, we learn the SHBC model on the constrained FERET face dataset and test them on the unconstrained LFW dataset. Though, strictly speaking, our experiments do not follow the LFW protocols because of the external datasets are used for training model, **but the effectiveness and generalization ability of our SHBC method is more convincing when it achieves the performance as well as LFW-based learning methods.** Similar to [26], we align and crop each image into (a) 150 × 130 with contour, (b) 150 × 130 without contour and (c) 128 × 128 without contour using different similarity transformations. The aligned and cropped face image with different similarity transformations can provide different global/local facial information. Figure 11 shows aligned and cropped sample images. The parameters of our SHBC are the same as those used on the above experiments. At last, the WPCA is applied to reducing the feature dimension of each image into 1,100 and the cosine metric is used to compute the similarity. Table 6 and Fig. 12 show the AUC percent and ROC curve of our SHBC methods and other the state-of-the-art methods

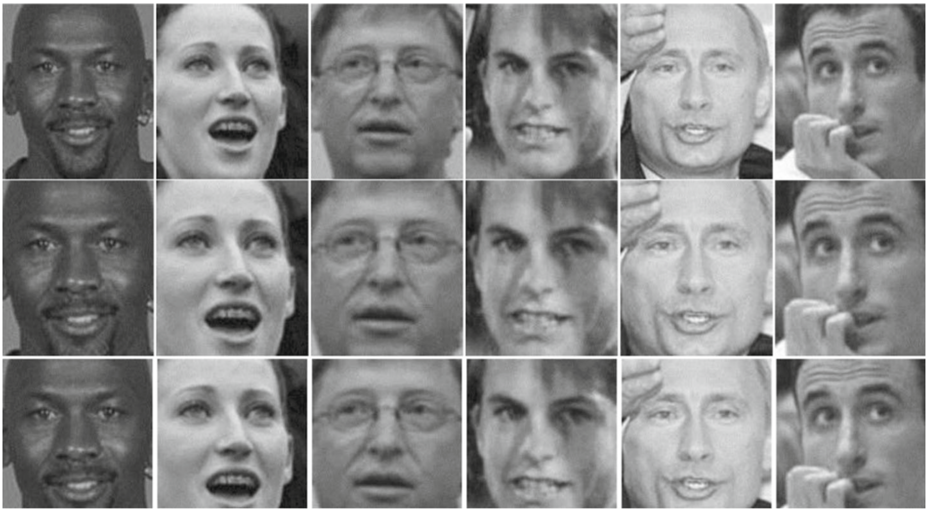


Fig. 11 Aligned and cropped face examples from the LFW dataset. The *top row* is (**a**) 150 × 130 with contour which can encodes both facial contour and components, and the *second row* is (**b**) 150 × 130 without contour which encodes facial components, and the *third row* is (**c**) 128 × 128 without contour which encodes facial components and complementary information

Table 6 The AUC (%) Comparisons with the State-of-the-art Methods on LFW under the Unsupervised Setting

Methods	AUC	Year
SIFT [38]*	54.07	2002
LBP [1]*	75.47	2006
LARK [36]*	78.30	2011
LHS [37]*	81.07	2012
LQP [17]*	87.00	2012
MRF-MLBP [4]*	89.94	2013
DFD [22]*	83.70	2014
CBFD [26]*	88.89	2015
JFL [25]*	91.03	2015
SHBC (a)	88.54	–
SHBC (b)	88.68	–
SHBC (c)	88.66	–
S-SHBC (a)	87.96	–
S-SHBC (b)	88.33	–
S-SHBC (c)	88.23	–

*Note that the results are from the original papers

under the unsupervised setting on the LFW dataset, where *SHBC (a)* and *S-SHBC (a)* indicates the result obtained on dataset *(a)* by using the *SHBC* and *S-SHBC* model, respectively. The rest results can be indicated in the same manner. In order to make Fig. 12 concise, we only plot the ROC curves which are published on the LFW website.[1] It is observed that our method achieves similar performance with the state-of-the-art descriptors like CBFD, MRF-MLBP and JFL, and obviously outperforms LARK, LHS, LQP and DFD. In addition, the performances of our proposed SHBC methods are better than other methods no matter what face images with different similarity transformations are used. Therefore, the SHBC method can efficiently extract the discriminative and robust feature representations by encoding either facial contour information or facial components information. **Although the intra/inter-class variabilities of FERET are very different from those on LFW, our SHBC model learned from constrained dataset can still work well in the unconstrained dataset.** It is enough to prove that our SHBC and S-SHBC have excellent generalization ability and it can be deployed into real scenarios.

5.4 Evaluation on PaSC

The experiment in the Section 5.3 can be considered as an evaluation of our methods' generalization ability in uncontrolled scenarios. In this section, we next evaluate the effectiveness of our SHBC method in real applications (i.e., both learned and tested on uncontrolled images). The Point-and-Shoot Cameras (PaSC) dataset consists of 9,376 still images from

[1] The results of other methods in this figure are cited from http://vis-www.cs.umass.edu/lfw/results.html

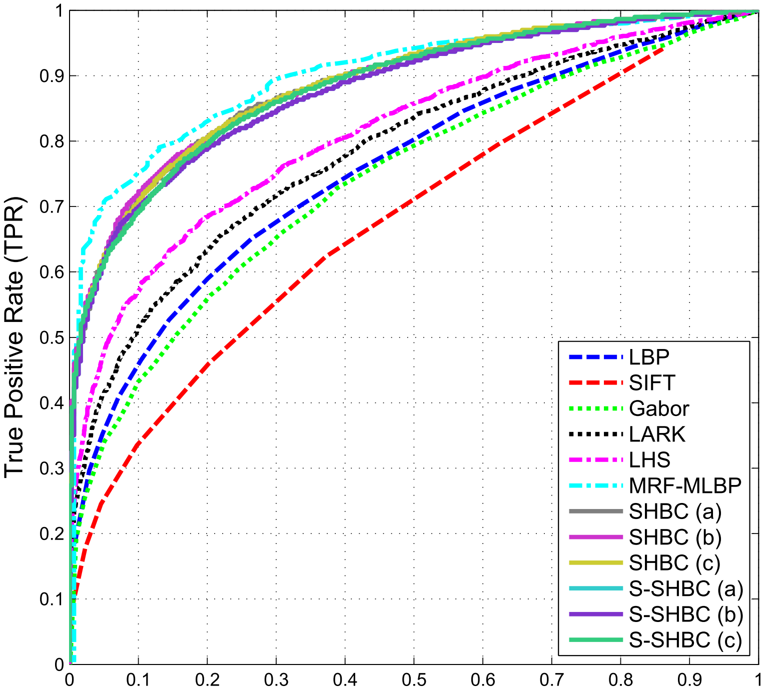


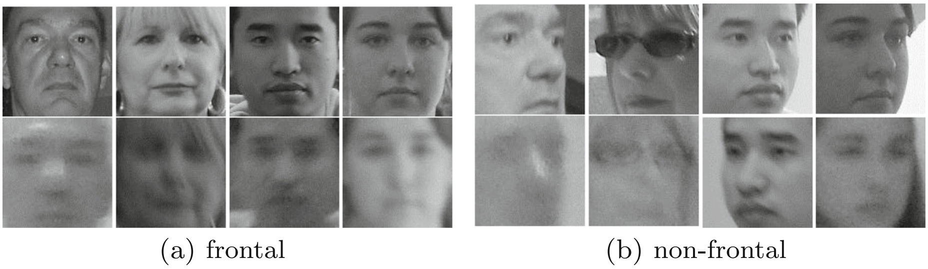
Fig. 12 The ROC curves of different methods on LFW under the unsupervised setting

293 subjects balanced with respect to distance, alternative sensors, poses and varying location. There are 4,688 images in both *query set* and *target set*. We align and crop each image into 128 × 128 pixels according to the eye coordinates provided by the PaSC database website.[2] Figure 13 shows some aligned and cropped face images from PaSC dataset. There are some complicated intra-class variabilities in PaSC dataset, such as pose, lighting, expression, motion blur and poor focus.

The parameters of our SHBC method are the same as those used on the LFW experiments. We performed feature learning process on the *target sets*. We follow the standard evaluation protocol where each image pairs are formed by taking one image from the query set and the other from the target set, so that the similarity scores for all image pairs in the test set are computed. By looking at image pairs of interest, two ROC curves can be plotted, one for frontal image pairs only and one that combines both frontal and non-frontal images.

We compare our SHBC method with CBFD, Local Phase Quantization (LPQ), Binarized Statistical Image Features (BSIF) and Randomized Intraclass-Distance Minimizing Binary Codes (RIDMBC), the conventional LBP, LRPCA and CohortLDA is used as baseline methods, where CBFD, LPQ, BSIF and RIDMBC are the state-of-the-art binary learning

[2] The eye coordinates of each face image can be found at http://www.cs.colostate.edu/~vision/pasc/index.php


(a) frontal

(b) non-frontal

Fig. 13 Aligned and cropped face examples from the PaSC dataset. **a** shows frontal or near-frontal face images and (**b**) shows non-frontal face images

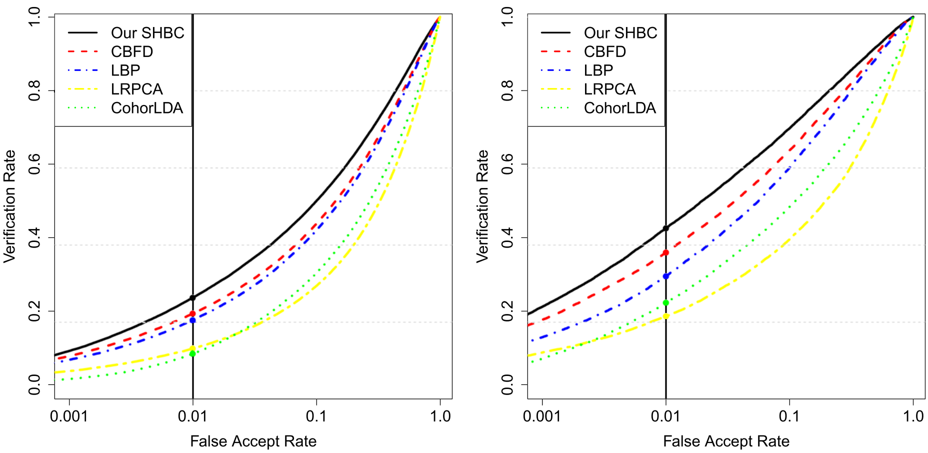
method. Since the structures and information for local face regions are different, for local binary learning method (i.e., LBP, CBFD and our IQBC), each image is divided into 8 × 8 non-overlapping blocks and the size of each block is 16 × 16. Then, we extract a 59-dimensional uniform LBP feature, 500-dimensional CBFD feature and 500-dimensional SHBC feature for each local block. At last, the concatenated feature vectors are reduced into 500-dimensional feature by WPCA as the final representation. Table 7 tabulates the verification rate at FAR=0.01 for all images and only frontal images. Figure 14 shows the ROC curves of some descriptors for all images and frontal image, respectively. In order to illustrate results clearly, we only plot the ROC curves of some representative descriptors. As can seen, our proposed SHBC methods significantly outperform the other state-of-the-art methods and baseline methods. Specially, our SHBC improves the CBFD by about 4.3 and 6.6 percent verification rate on all and frontal test set, respectively. The objective function in our SHBC model is effective to extract discriminative and data-adaptive feature representations, therefore, our SHBC descriptor can achieve the best verification rate compared to other descriptors. Our SHBC descriptor can not only learn discriminative feature representations on constrained scenarios, but also show excellent generalization ability and robustness when faced with real unconstrained scenarios. We can expected that our SHBC model can be deployed into real scenarios if a large enough unconstrained face database is used to train our model.

Table 7 Verification Rate (%) at FAR=0.01 on PaSC dataset for all images and frontal images

Method	All	Frontal	Year
LBP [1]	17.6	29.6	2006
LRPCA [33]*	10.0	19.0	2011
CohortLDA [27]*	8.0	22.0	2012
LPQ [35] *	13.2	23.1	2012
BSIF [52]*	14.3	24.9	2014
RIDMBC [56]*	14.9	23.5	2014
CBFD [26]	19.4	36.0	2015
SHBC	**23.7**	**42.6**	–

*Note that the results are from the original papers

The bold values denote the best performance on the subsets



(a) all

(b) frontal

Fig. 14 The ROC curves of different methods on the PaSC dataset for all and frontal images scenarios respectively

6 Conclusion and future work

This paper proposes a learning-based descriptor called spherical hashing binary code for face representation and recognition in the data-driven way. Different from the previous face descriptor, our SHBC method learn feature representation from raw data and our proposed descriptors are binary feature descriptor. Therefore, it can extract more discriminative and robust feature than previous read-valued learning methods from raw pixel patches. According to whether label information is used or not, we propose unsupervised SHBC and supervised S-SHBC in Sections 3 and 4, respectively. Extensive experiments are conducted on constrained face datasets, such as FERET and CAS-PEAL-R1, and unconstrained datasets, such as LFW and PaSC. The experiment results show that the proposed SHBC and S-SHBC can extract discriminative feature and have good generalization under different scenarios.

There are two possible research directions for our future work:

1. Our SHBC methods show excellent generalized power under unconstrained scenarios. Therefore, we can train them in a large scale dataset that collects sufficiently inter-class and intra-class variations, and apply them to solve some problems of unconstrained face recognition scenarios in real life.
2. Since our SHBC and S-SHBC are general learning-based methods, ideally, it can learn discriminative feature representations of various visual tasks as long as extensive training samples are provided. Therefore, we expect that they can further demonstrate its effectiveness on other computer vision tasks, such as texture recognition and expression recognition.

Acknowledgments The work presented in this paper was supported by the National Natural Science Foundation of China (Grants No. NSFC-61402046, NSFC-61170176), Fund for Beijing University of Posts and Telecommunications (No.2013XZ10, 2013XD-04), Fund for the Doctoral Program of Higher Education of China (Grants No.20120005110002).

References

1. Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: application to face recognition. IEEE Trans Pattern Anal Mach Intell 28(12):2037–2041. doi:10.1109/TPAMI.2006.244
2. Andoni A, Indyk P (2006) Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In: 47th Annual IEEE symposium on foundations of computer science, 2006. FOCS'06. IEEE, pp 459–468
3. Andoni A, Indyk P (2008) Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. Commun ACM 51(1):117–122. doi:10.1145/1327452.1327494
4. Arashloo SR, Kittler J (2013) Efficient processing of mrfs for unconstrained-pose face recognition. In: 2013 IEEE Sixth international conference on biometrics: theory, applications and systems (BTAS). IEEE, pp 1–8
5. Bartlett M, Movellan JR, Sejnowski T (2002) Face recognition by independent component analysis. IEEE Trans Neural Netw 13(6):1450–1464. doi:10.1109/TNN.2002.804287
6. Belhumeur P, Hespanha J, Kriegman D (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720. doi:10.1109/34.598228
7. Beveridge JR, Phillips J, Bolme DS, Draper B, Givens GH, Lui YM, Teli MN, Zhang H, Scruggs WT, Bowyer KW et al (2013) The challenge of face recognition from digital point-and-shoot cameras. In: 2013 IEEE Sixth international conference on biometrics: theory, applications and systems (BTAS). IEEE, pp 1–8
8. Chakraborty S, Singh SK, Chakraborty P (2015) Local directional gradient pattern: a local descriptor for face recognition. Multimed Tools Appl:1–16
9. Chan TH, Jia K, Gao S, Lu J, Zeng Z, Ma Y (2014) Pcanet: a simple deep learning baseline for image classification? arXiv preprint arXiv:14043606
10. Fan KC, Hung TY (2014) A novel local pattern descriptor-local vector pattern in high-order derivative space for face recognition. IEEE Trans Image Process 23(7):2877–2891. doi:10.1109/TIP.2014.2321495
11. Gao W, Cao B, Shan S, Chen X, Zhou D, Zhang X, Zhao D (2008) The cas-peal large-scale chinese face database and baseline evaluations. IEEE Trans Syst Man Cybern Part A: Syst and Humans 38(1):149–161
12. Gong Y, Lazebnik S (2011) Iterative quantization: a procrustean approach to learning binary codes. In: 2011 IEEE conference on computer vision and pattern recognition (CVPR), pp 817–824. doi:10.1109/CVPR.2011.5995432
13. Gong Y, Kumar S, Verma V, Lazebnik S (2012) Angular quantization-based binary codes for fast similarity search. In: Advances in neural information processing systems, pp 1196–1204
14. Gong Y, Lazebnik S, Gordo A, Perronnin F (2013) Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval. IEEE Trans Pattern Anal Mach Intell 35(12):2916–2929. doi:10.1109/TPAMI.2012.193
15. Guo Z, Zhang L, Zhang D (2010) A completed modeling of local binary pattern operator for texture classification. IEEE Trans Image Process 19(6):1657–1663
16. Huang GB, Ramesh M, Berg T, Learned-Miller E (2007) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Tech. rep. Technical Report 07-49. University of Massachusetts, Amherst
17. Hussain SU, Napoléon T, Jurie F (2012) Face recognition using local quantized patterns. In: British machive vision conference, pp 11–pages
18. Kan M, Shan S, Chang H, Chen X (2014) Stacked progressive auto-encoders (spae) for face recognition across poses. In: 2014 IEEE conference on computer vision and pattern recognition (CVPR), pp 1883–1890. doi:10.1109/CVPR.2014.243
19. Kulis B, Grauman K (2012) Kernelized locality-sensitive hashing. IEEE Trans Pattern Anal Mach Intell 34(6):1092–1104. doi:10.1109/TPAMI.2011.219
20. Lei Z, Li SZ, Chu R, Zhu X (2007) Face recognition with local gabor textons. In: Advances in biometrics. Springer, pp 49–57
21. Lei Z, Liao S, Pietikainen M, Li S (2011) Face recognition by exploring information jointly in space, scale and orientation. IEEE Trans Image Process 20(1):247–256. doi:10.1109/TIP.2010.2060207
22. Lei Z, Pietikainen M, Li SZ (2014) Learning discriminant face descriptor. IEEE Trans Pattern Anal Mach Intell 36(2):289–302

23. Liu C, Wechsler H (2002) Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. IEEE Trans Image Process 11(4):467–476. doi:10.1109/TIP.2002.999679
24. Liu W, Wang J, Kumar S, Chang SF (2011) Hashing with graphs. In: Proceedings of the 28th international conference on machine learning (ICML-11), pp 1–8
25. Lu J, Liong V, Wang G, Moulin P (2015a) Joint feature learning for face recognition. IEEE Trans Inform Forens Secur 10(7):1371–1383. doi:10.1109/TIFS.2015.2408431
26. Lu J, Liong V, Zhou X, Zhou J (2015b) Learning compact binary face descriptor for face recognition. IEEE Trans Pattern Anal Mach Intell PP(99):1–1. doi:10.1109/TPAMI.2015.2408359
27. Lui YM, Bolme D, Phillips PJ, Beveridge JR, Draper B et al (2012) Preliminary studies on the good, the bad, and the ugly face recognition challenge problem. In: 2012 IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW). IEEE, pp 9–16
28. Maturana D, Mery D, Soto A (2011a) Learning discriminative local binary patterns for face recognition. In: 2011 IEEE International conference on automatic face & gesture recognition and workshops (FG 2011). IEEE, pp 470–475
29. Maturana D, Mery D, Soto l (2011b) Face recognition with decision tree-based local binary patterns. In: Kimmel R, Klette R, Sugimoto A (eds) Computer vision ACCV 2010, lecture notes in computer science, vol 6495. Springer, Berlin Heidelberg, pp 618–629. doi:10.1007/978-3-642-19282-1_49
30. Meng X, Shan S, Chen X, Gao W (2006) Local visual primitives (lvp) for face modelling and recognition. In: 18th International conference on pattern recognition, 2006. ICPR 2006. IEEE, vol 2, pp 536–539
31. Meyers E, Wolf L (2008) Using biologically inspired features for face processing. Int J Comput Vis 76(1):93–104
32. Phillips PJ, Moon H, Rizvi S, Rauss PJ et al (2000) The feret evaluation methodology for face-recognition algorithms. IEEE Trans Pattern Anal Mach Intell 22(10):1090–1104
33. Phillips PJ, Beveridge JR, Draper B, Givens G, Toole AJ, Bolme DS, Dunlop J, Lui YM, Sahibzada H, Weimer S et al (2011) An introduction to the good, the bad, & the ugly face recognition challenge problem. In: 2011 IEEE International conference on automatic face & gesture recognition and workshops (FG 2011). IEEE, pp 346–353
34. Raginsky M, Lazebnik S (2009) Locality-sensitive binary codes from shift-invariant kernels. In: Bengio Y, Schuurmans D, Lafferty J, Williams C, Culotta A (eds) Advances in neural information processing systems, vol 22. Curran Associates Inc., pp 1509–1517
35. Rahtu E, Heikkilä J, Ojansivu V, Ahonen T (2012) Local phase quantization for blur-insensitive image analysis. Image Vis Comput 30(8):501–512
36. Seo HJ, Milanfar P (2011) Face verification using the lark representation. IEEE Trans Inform Forens Secur 6(4):1275–1286
37. Sharma G, ul Hussain S, Jurie F (2012) Local higher-order statistics (lhs) for texture categorization and facial analysis. In: Computer vision–ECCV 2012. Springer, pp 1–12
38. Singh R, Om H (2013) An overview of face recognition in an unconstrained environment. In: 2013 IEEE Second international conference on image information processing (ICIIP), pp 672–677. doi:10.1109/ICIIP.2013.6707679
39. Sun Y, Chen Y, Wang X, Tang X (2014) Deep learning face representation by joint identification-verification. In: Advances in neural information processing systems, pp 1988–1996
40. Taigman Y, Ming Y, Ranzato M, Wolf L (2014) Deepface: closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on computer vision and pattern recognition (CVPR), pp 1701–1708. doi:10.1109/CVPR.2014.220
41. Tian L, Fan C, Ming Y, Jin Y (2015) Stacked pca network (spcanet): an effective deep learning for face recognition. In: 2015 IEEE International conference on digital signal processing (DSP). IEEE, pp 1039–1043
42. Turk M, Pentland A (1991) Face recognition using eigenfaces. In: IEEE Computer society conference on computer vision and pattern recognition, 1991. Proceedings CVPR '91., pp 586–591. doi:10.1109/CVPR.1991.139758
43. Vu NS, Caplier A (2012) Enhanced patterns of oriented edge magnitudes for face recognition and image matching. IEEE Trans Image Process 21(3):1352–1365
44. Wang N, Li Q, El-Latif AAA, Peng J, Niu X (2014) An enhanced thermal face recognition method based on multiscale complex fusion for gabor coefficients. Multimed Tools Appl 72(3):2339–2358

45. Weiss Y, Torralba A, Fergus R (2009) Spectral hashing. In: Koller D, Schuurmans D, Bengio Y, Bottou L (eds) Advances in neural information processing systems, vol 21. Curran Associates Inc., pp 1753–1760. http://papers.nips.cc/paper/3383-spectral-hashing.pdf
46. Werghi N, Berretti S, Del Bimbo A (2015) The mesh-lbp: a framework for extracting local binary patterns from discrete manifolds. IEEE Trans Image Process 24(1):220–235. doi:10.1109/TIP.2014.2370253
47. Xia Y, He K, Kohli P, Sun J (2015) Sparse projections for high-dimensional binary codes. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3332–3339
48. Xie S, Shan S, Chen X, Meng X, Gao W (2009) Learned local gabor patterns for face representation and recognition. Signal Process 89(12):2333–2344
49. Xie S, Shan S, Chen X, Chen J (2010) Fusing local patterns of gabor magnitude and phase for face recognition. IEEE Trans Image Process 19(5):1349–1361. doi:10.1109/TIP.2010.2041397
50. Yang BQ, Zhang T, Gu CC, Wu KJ, Guan XP (2015) A novel face recognition method based on iwld and iwbc. Multimed Tools Appl:1–24
51. Yi S, Xiaogang W, Xiaoou T (2014) Deep learning face representation from predicting 10,000 classes. In: 2014 IEEE Conference on computer vision and pattern recognition (CVPR), pp 1891–1898. doi:10.1109/CVPR.2014.244
52. Ylioinas J, Hadid A, Kannala J, Pietikainen M (2014) An in-depth examination of local binary descriptors in unconstrained face recognition. In: 2014 22nd International conference on pattern recognition (ICPR). IEEE, pp 4471–4476
53. Yu FX, Kumar S, Gong Y, Chang SF (2014) Circulant binary embedding. arXiv preprint arXiv:14053162
54. Zhang B, Shan S, Chen X, Gao W (2007) Histogram of gabor phase patterns (hgpp): a novel object representation approach for face recognition. IEEE Trans Image Process 16(1):57–68. doi:10.1109/TIP.2006.884956
55. Zhang B, Gao Y, Zhao S, Liu J (2010) Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. IEEE Trans Image Process 19(2):533–544
56. Zhang H, Beveridge J, Mo Q, Draper B, Phillips P (2014) Randomized intraclass-distance minimizing binary codes for face recognition. In: 2014 IEEE international joint conference on biometrics (IJCB), pp 1–8. doi:10.1109/BTAS.2014.6996258
57. Zhang W, Shan S, Gao W, Chen X, Zhang H (2005) Local gabor binary pattern histogram sequence (lgbphs): a novel non-statistical model for face representation and recognition. In: Tenth IEEE international conference on computer vision, 2005. ICCV 2005, vol 1, pp 786–791. doi:10.1109/ICCV.2005.147

Lei Tian he received the M.S. degree in the Department of Electronic Engineering from Beijing University of Posts and Telecommunications, China, in 2015. He is currently working toward the Ph.D degree in the Beijing Key Laboratory of Work Safety Intelligent Monitoring, the Department of Electronic Engineering, Beijing University of Posts and Telecommunications. His research interests include computer vision, pattern recognition, deep learning and face recognition in particular. He has authored 4 scientific papers in international journals and conferences about these areas.

Chunxiao Fan she is currently a professor and the director of Center for information electronic and intelligence system. She served as a member of ISO/IEC JTC1/SC6 WG9, ASN.1 (since 2006) and Chinese Sensor network working group. She also was elevated to evaluation expert of Beijing Scientific and Technical Academy Awards. Her research interests include Heterogeneous media data analysis, Internet of Things, data mining, communication software and so on. In recent years, she is director of several Nation Science Foundation Project. She has published more than 30 papers in international journals and conferences, authored and edited three books and has authorized several patent for invention.

Yue Ming she received the B.S. degree in Communication Engineering, and the M.S. degree in Human-Computer Interaction Engineering, and Ph.D degree in Signal and Information Processing from Beijing Jiaotong University, China, in 2006, 2008, and 2013, respectively. She worked as a visiting scholar in Carnegie Mellon University, U.S., between 2010 and 2011. Since 2013, she has been working as a faculty member at Beijing University of Posts and Telecommunications. Her research interests are in the areas of biometrics, computer vision, computer graphics, information retrieval, pattern recognition, etc. She has authored more than 40 scientific papers in these areas.