Context-Aware Local Binary Feature Learning for Face Recognition

Yueqi Duan, Jiwen Lu, *Senior Member, IEEE*, Jianjiang Feng, *Member, IEEE*, and Jie Zhou, *Senior Member, IEEE*

Abstract—In this paper, we propose a context-aware local binary feature learning (CA-LBFL) method for face recognition. Unlike existing learning-based local face descriptors such as discriminant face descriptor (DFD) and compact binary face descriptor (CBFD) which learn each feature code individually, our CA-LBFL exploits the contextual information of adjacent bits by constraining the number of shifts from different binary bits, so that more robust information can be exploited for face representation. Given a face image, we first extract pixel difference vectors (PDV) in local patches, and learn a discriminative mapping in an unsupervised manner to project each pixel difference vector into a context-aware binary vector. Then, we perform clustering on the learned binary codes to construct a codebook, and extract a histogram feature for each face image with the learned codebook as the final representation. In order to exploit local information from different scales, we propose a context-aware local binary multi-scale feature learning (CA-LBMFL) method to jointly learn multiple projection matrices for face representation. To make the proposed methods applicable for heterogeneous face recognition, we present a coupled CA-LBFL (C-CA-LBFL) method and a coupled CA-LBMFL (C-CA-LBMFL) method to reduce the modality gap of corresponding heterogeneous faces in the feature level, respectively. Extensive experimental results on four widely used face datasets clearly show that our methods outperform most state-of-the-art face descriptors.

Index Terms—Face recognition, binary feature learning, context-aware, multi-feature learning, heterogeneous face matching

1 Introduction

Face recognition has attracted much attention in computer vision and numerous face recognition methods have been proposed over the past three decades [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. As a representative pattern recognition problem, there are two main procedures in a practical face recognition system: face representation and face matching. Face representation aims to extract discriminative features to separate face images of different persons, and face matching is to design effective classifiers to recognize different persons.

A variety of face representation methods have been proposed in recent years [1], [2], [3], [8], [13], [14], and they can be mainly classified into two categories: holistic feature representation [2], [5] and local feature representation [1], [3], [8], [13]. Representative holistic features are principal component analysis (PCA) [5], linear discriminant analysis (LDA) [2], and their variations [2], [5]. Representative local features include local binary patterns (LBP) [1], Gabor descriptor [3], discriminant face descriptor (DFD) [8] and compact binary face descriptor (CBFD) [13]. Generally, local features achieve better performance than holistic features due to their stableness and robustness to local changes in feature description [13], [15], [16].

Most existing local feature descriptors are hand-crafted [1], [15], [16], [17], which usually require strong prior knowledge and are heuristics. While learning-based methods such as DFD and CBFD learn feature representations from raw pixels directly, they only learn each feature code individually and are more susceptible to noise. Contextual information is an effective manner to address the limitation of such unstableness because context provides strong prior knowledge, which enhances the robustness and stableness of various visual analysis tasks such as video understanding [18], object detection [19] and visual recognition [20]. Inspired by the fact that contextual information can provide effective cues to improve the robustness of binary codes, we propose a context-aware local binary feature learning (CA-LBFL) method for face recognition, which learns context-aware binary codes directly from raw pixels for face representation. Compared with existing feature learning methods which learn feature codes separately, our CA-LBFL exploits the contextual information of adjacent bits by limiting the number of bitwise changes in each descriptor, and obtains more robust local binary features. First, we extract pixel difference vectors (PDV) for each face image, and learn a mapping matrix to project each PDV into a context-aware binary vector. Then, we learn a codebook by clustering from all binary codes, and construct a histogram feature for each face image with the codebook as the final representation. Fig. 1 illustrates the pipeline of our proposed approach. As the values from different scales are simply

• *The authors are with the Department of Automation, State Key Lab of Intelligent Technologies and Systems, and Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, Beijing 100084, China. E-mail: duanyq14@mails.tsinghua.edu.cn, {lujiwen, jfeng, jzhou}@tsinghua.edu.cn.*

Manuscript received 7 Mar. 2016; revised 2 May 2017; accepted 25 May 2017. Date of publication 30 May 2017; date of current version 10 Apr. 2018. (Corresponding author: Jiwen Lu.)
Recommended for acceptance by T. Darell, C. Lampert, N. Sebe, Y. Wu, and Y. Yan. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2017.2710183

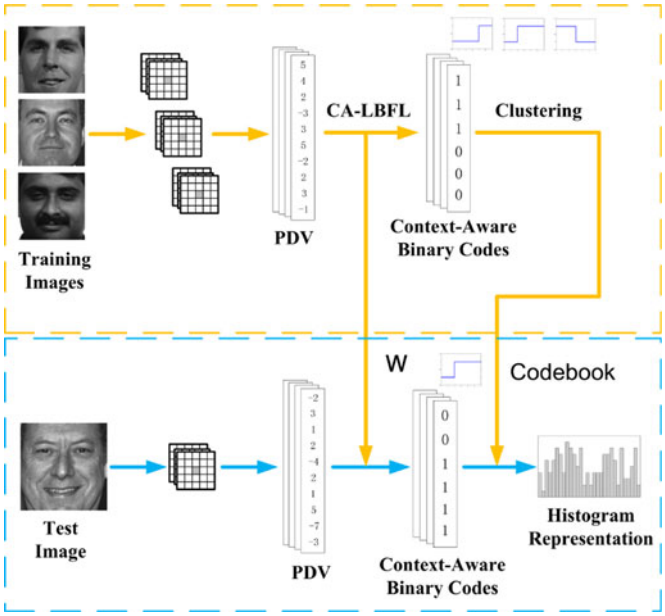


Fig. 1. The flow-chart of our proposed CA-LBFL approach for face representation. For each training image, we first extract pixel difference vectors (PDV) and learn a discriminative mapping $\mathbf{W}$ to project each PDV into context-aware binary codes, where adjacent bits are enforced as equal as possible to enhance the robustness of the descriptor. Then, a codebook is learned by clustering for feature encoding. For each test image, the PDVs are first extracted and then projected into context-aware binary codes using the learned feature mapping. Finally, a histogram feature descriptor is extracted from binary codes with the learned codebook.

concatenated into PDV, the diversity of scales cannot be well exploited. To address this, we present a context-aware local binary multi-scale feature learning (CA-LBMFL) method to jointly learn multiple projection matrices to make better use of multiple scale local information. Moreover, we propose a coupled CA-LBFL (C-CA-LBFL) method and a coupled CA-LBMFL (C-CA-LBMFL) method which minimize the modality gap of corresponding heterogeneous faces in the feature level to adapt our methods to heterogeneous face recognition. Extensive experimental results on LFW, YouTubeFace (YTF), FERET and CASIA NIR-VIS 2.0 show that our methods outperform most state-of-the-art face descriptors.

The main contributions of this work are summarized as follows:

1) We propose an unsupervised local feature learning method to learn context-aware binary descriptors for face representation. CA-LBFL exploits contextual information of the adjacent binary bits, which provides effective prior knowledge to learn robust binary feature representations. With the contextual information, the learned local binary features are more stable to local changes and deliver stronger discriminative power.
2) We apply a joint learning method to learn multiple projection for feature mapping. The proposed CA-LBMFL method exploits the specific characteristic from different scales as well as the interactions of different projection of matrices, which can make better use of multiple local scale information.
3) We further develop coupled learning methods based on CA-LBFL and CA-LBMFL for heterogeneous face matching. The coupled methods learn pairs of hash functions for different modalities simultaneously, which minimize the modality gap of heterogeneous faces in the feature level.

2 Background

In this section, we briefly review three related topics: 1) face representation, 2) feature learning, and 3) binary feature descriptor.

2.1 Face Representation

Face representation mainly includes two categories: homogeneous face representation and heterogeneous face representation. Homogeneous face representation aims to recognize faces from the same modality, while heterogeneous face representation matches faces from different sources, such as visible photos to near infrared images or sketches.

Existing homogeneous face representation methods can be mainly classified into two categories: holistic feature representation [2], [5] and local feature representation [1], [3], [8], [13]. Holistic features learn a feature subspace to preserve the statistical information of face images, where PCA [5] and LDA [2] are representative such methods. Local features describe the structure pattern of each local patch rather than the whole image, and combine the statistics of all patches to build a concatenated feature vector. Local feature representation methods can be divided into hand-crafted and learning-base. Hand-crafted methods such as LBP [1] and Gabor wavelets [3] compute the gradient or texture information within local regions first and then generate a concatenated feature vector for face representation. However, these features usually require strong prior knowledge to engineer them. Learning-based methods such as DFD [8] and CBFD [13] learn distinctive local features in a data-driven way.

Heterogeneous face representation suffers from large modal discrepancies and it is desirable to design cross-modal models which are robust to the intra-modality differences. Existing heterogeneous face representation methods mainly contains three categories: image synthesis [21], [22], modality-invariant feature extraction [23], [24] and common space projection [25], [26]. Image synthesis approaches transform faces of one modality into another, so that heterogeneous facial images can be directly compared. Representative methods include face sketch synthesis with embedded hidden Markov model (E-HMM) [21] and face identity-preserving (FIP) features [22]. Modality-invariant feature extraction approaches extract local features which are robust to modalities, such as histogram of averaged oriented gradients (HAOG) [23] and graphical heterogeneous face recognition (G-HFR) [24]. However, both image synthesis and modality-invariant feature extraction approaches are modality-specific. Common space projection methods learn a common subspace to minimize the modal differences. For example, Yi et al. [25] learned a canonical correlation analysis (CCA) based projection. Mignon and Jurie [26] presented a cross modal metric learning (CMML) approach by learning a common subspace. Different from modality-specific heterogeneous face recognition approaches, our C-CA-LBFL and C-CA-LBMFL learn a common subspace in an unsupervised manner, which are widely applicable to various heterogeneous face recognition tasks.

2.2 Feature Learning

There have been extensive work on feature learning in recent years [27], [28], [29], [30], [31], [32], and representative feature learning models include sparse auto-encoders [27], denoising auto-encoders [28], restricted Boltzman machine [29], convolutional neural networks [30], independent subspace analysis [31], and reconstructio independent component analysis [32]. Recently, feature learning based methods have achieved reasonably good performance in many face recognition systems [8], [33], [34], [35]. For example, Sun et al. [33] proposed a deep hidden identity features (DeepID) method through deep convolutional neural networks. Hussain et al. [34] presented a local quantized pattern (LQP) method by modifying the LBP method with a learned coding strategy. Cao et al. [35] proposed learning-based (LE) feature representation method by applying the bag-of-word (BoW) framework. Lei et al. [8] presented a discriminant face descriptor method by learning an image filter using the LDA criterion to obtain LBP-like features. Lu et al. [13] proposed a compact binary feature descriptor by learning a hashing filter to project each image patch into a compact binary vector in an unsupervised manner. They also presented a simultaneous local binary feature learning and encoding (SLBFLE) [14] approach to simultaneous learn the projection matrix and the dictionary. However, both CBFD and SLBFLE only exploit the relationship of binary bits at the same position, while the proposed CA-LBFL investigates the contextual information within each binary descriptor.

2.3 Binary Feature Descriptor

Recently, binary feature descriptors have received increasing interest due to their efficiency of storing and matching in computer vision. Earlier binary descriptors include binary robust independent elementary feature (BRIEF) [36], oriented FAST and rotated BRIEF (ORB) [37], binary robust invariant scalable keypoint (BRISK) [38] and fast retina keypoint (FREAK) [39]. BRIEF computes binary vectors directly by simple binary tests between pixels in a smoothed image patch. ORB improves BRIEF by employing scale pyramids and orientation operators to obtain scale and rotation invariance. BRISK shares the similar purpose as ORB by leveraging a circular sampling pattern. FREAK uses the retinal sampling grid for fast computing and matching inspired by the human visual system, retina. However, the performance of these methods is not powerful enough because raw intensity comparisons are susceptible to scale and transformation. To address this, several learning-based methods [40], [41], [42], [43], [44], [45] have been proposed in recent years. For example, Trzcinski et al. [43] presented D-BRIEF method to learn discriminative projections by encoding similarity relationships. They also applied boosting to learn hash functions in BinBoost [44]. Balntas et al. [40] proposed a binary online learned descriptor (BOLD) maximize the inter-class distances as well as minimize the intra-class distances of binary codes, respectively.

3 Context-Aware Local Binary Feature Learning

In this section, we first present the proposed CA-LBFL method, and then introduce how to use CA-LBFL for face representation.

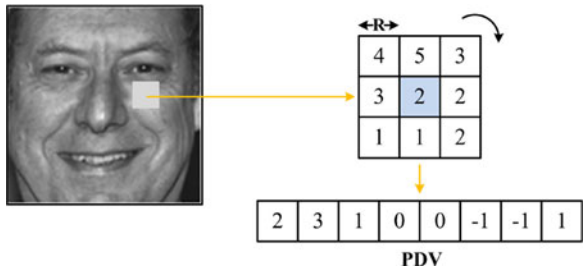


Fig. 2. An illustration to show how to extract a pixel difference vector (PDV) from the original face image in our approach. Given any pixel in the image, we first compute the differences between the central pixel and its $(2R+1)\times(2R+1)$ neighboring pixels, where R is the parameter to set the neighborhood size. Then, these differences are aligned as a vector which becomes the PDV feature of the pixel. R is selected as 1 in this figure for easy illustration, so that the PDV is a 8-dimensional vector as there are eight neighboring pixels selected.

3.1 CA-LBFL

Let $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N]$ be the N samples of the training set, where $\mathbf{x}_n\in\mathbb{R}^d$ $(1\le n\le N)$ is the nth pixel difference vector obtained from an original image. Fig. 2 illustrates the approach to extract a PDV from the given face patch. In this work, we utilize raw pixels rather than HSV to compute PDV. More specifically, we use grayscale values of pixels which are the means of the three channels of RGB. PDV is able to encode important visual patterns such as lines and edges as it describes the changes of pixel values by measuring the differences between the central pixel and neighboring pixels. In our method, we learn K hash functions to map and quantize each $\mathbf{x}_n$ into a binary vector $\mathbf{b}_n=[b_{1n},\ldots,b_{Kn}]^T\in\{0,1\}^{K\times 1}$ to obtain context-aware binary codes. Let $\mathbf{w}_k\in\mathbb{R}^d$ be the projection vector for the kth function, and the kth binary code b_{kn} of $\mathbf{x}_n$ can be computed as

$$b_{kn}=0.5\times(\mathrm{sgn}(\mathbf{w}_k^T\mathbf{x}_n)+1), \tag{1}$$

where $\mathrm{sgn}(v)$ equals to 1 if $v\ge 0$ and -1 otherwise.

In order to make our binary codes context-aware, adjacent bits should be as equal as possible, e.g., "11100000" and "00111000". However, this restraint may enforce each learned binary code to all-zeros or all-ones, which largely reduces the distinctiveness of binary codes. Therefore, a desirable binary code should be that there is only one shift between 0 and 1 in each binary vector. Bits in context-unaware binary codes are more likely to be affected by noise in face images because there is no limitation to prevent such bitwise changes, such as the $0/1$ shift of $\mathbf{B}$ in Fig. 3a. Context-aware learning partially solves this problem by limiting the sum of bitwise $0/1$ changes in each binary code and making the code more smooth. Fig. 3b illustrates that even if small changes affect the original code $\mathbf{X}$, the learned binary descriptor keeps stable as potential bitwise changes are intercepted due to the limits of contextual information. Context-aware is an important criterion to learn compact binary codes, as noise is suppressed due to the contextual information, and the robustness of the binary codes is improved without harm of distinctiveness.

Inspired by the above motivations, we formulate the following optimization objective function to binary codes for feature representation

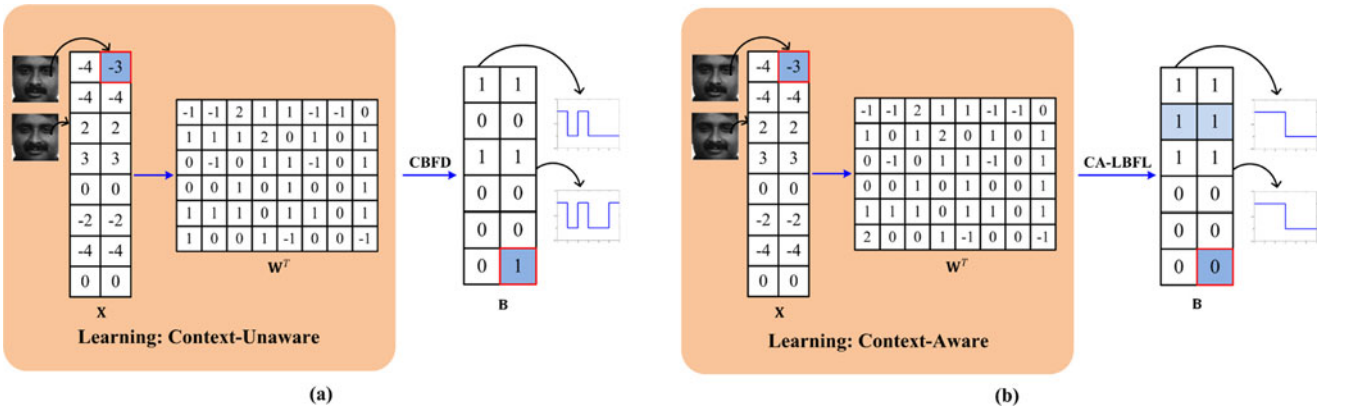


Fig. 3. Learning without contextual information in (a) CBFD and (b) our CA-LBFL. CA-LBFL learns context-aware binary codes that adjacent bits tend to be equal. We see that when face images are affected by variations such as varying poses, expressions, illuminations, occlusions, resolutions, and backgrounds, binary codes learned by our method still keep smooth and are more stable compared with context-unaware methods.

$$\begin{aligned}\min_{\mathbf{w}_k} J &= J_1 + \lambda_1 J_2 + \lambda_2 J_3 + \lambda_3 J_4 \\ &= \sum_{n=1}^{N}\left\|\sum_{k=1}^{K-1}\|b_{kn} - b_{(k+1)n}\|^2 - 1\right\|^2 \\ &\quad + \lambda_1 \sum_{n=1}^{N}\sum_{k=1}^{K}\left\|(b_{kn} - 0.5) - \mathbf{w}_k^T\mathbf{x}_n\right\|^2 \\ &\quad + \lambda_2 \sum_{k=1}^{K}\left\|\sum_{n=1}^{N}(b_{kn} - 0.5)\right\|^2 \\ &\quad - \lambda_3 \sum_{n=1}^{N}\sum_{k=1}^{K}\|b_{kn} - \mu_k\|^2,\end{aligned} \tag{2}$$

where N is the number of PDVs which are extracted from the original images, μ_k is the mean of the kth bit of all N PDVs, and λ_1, λ_2, λ_3 are three parameters to balance the weight of different terms.

In the first term J_1, the physical meaning of $\sum_{k=1}^{K-1} \| b_{kn} - b_{(k+1)n}\|^2$ is the sum of bitwise 0/1 changes in each binary vector. As aforementioned, to avoid making all the learned binary codes the same (all zeros or ones), we encourage the sum of bitwise changes to be one. Fig. 4 illustrates an example of calculating J_1. The minimization of J_1 makes the adjacent bits in the learned binary codes equal as possible, as well as avoiding the appearance of all zeros or ones, so that the contextual information is used and the codes are more robust to the noises. J_2 aims to reduce the quantization loss between the original features and the learned binary codes, which minimizes the loss of energy in the process of projection. J_3 aims to make each feature bit in the learned binary codes is evenly distributed, so that most information can be conveyed by each bit. J_4 is to maximize the variance of the learned binary codes to make the each projection vector as independent as possible.

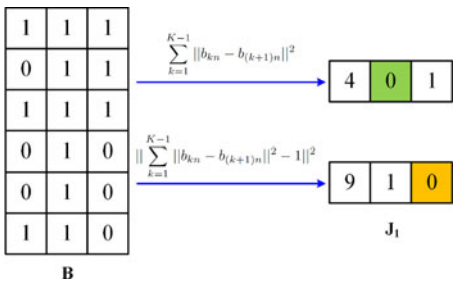


Fig. 4. An illustration of the physical meaning of J_1. The formula above represents the number of bitwise changes in each binary code which encourages all-zeros or all-ones descriptors, while the below one prefers one-shift binary descriptors and improves the diversity of learned binary codes.

Let $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_K] \in \mathbb{R}^{d\times K}$ be the projection matrix, and each sample $\mathbf{x}_n$ can be mapped into a binary vector as follows:

$$\mathbf{b}_n = 0.5 \times (\mathrm{sgn}(\mathbf{W}^T\mathbf{x}_n) + 1). \tag{3}$$

Then, (2) can be re-written into the matrix form as

$$\begin{aligned}\min_{\mathbf{W}} J &= J_1 + \lambda_1 J_2 + \lambda_2 J_3 + \lambda_3 J_4 \\ &= \mathrm{tr}(((\mathbf{AB})^T(\mathbf{AB}) - \mathbf{I}_N)^2) \\ &\quad + \lambda_1 ||(\mathbf{B} - 0.5) - \mathbf{W}^T\mathbf{X}||_F^2 \\ &\quad + \lambda_2 ||(\mathbf{B} - 0.5) \times \mathbf{1}^{N\times 1}||_F^2 \\ &\quad - \lambda_3 \mathrm{tr}((\mathbf{B} - \mathbf{U})^T(\mathbf{B} - \mathbf{U})),\end{aligned} \tag{4}$$

where $\mathbf{B} = 0.5 \times (\mathrm{sgn}(\mathbf{W}^T\mathbf{X} + 1) \in \{0, 1\}^{K\times N}$ is the matrix of all binary codes, $\mathbf{U} \in \mathbb{R}^{K\times N}$ is the mean matrix repeating the row vector of the mean of all binary bits, $\mathbf{I}_N$ is the identity matrix, and matrix $\mathbf{A} \in \{0, 1, -1\}^{(K-1)\times K}$ is designed to minimize the difference between adjacent bits in binary codes as follows:

$$a_{ij} = \begin{cases} 1, & i = j; \\ -1, & i = j - 1; \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

where a_{ij} is the element of the matrix $\mathbf{A}$, and i and j are the indices. Therefore, $\mathbf{AB}$ is the matrix that represents the differences between all the adjacent bits in learned binary codes, and the diagonal of $(\mathbf{AB})^T(\mathbf{AB})$ is the sum of bitwise changes of each learned binary code.

As the non-linear sgn(·) function makes (4) an NP-hard problem, we relax the sgn(·) function as its signed magnitude [46]. Thus, J_1 can be rewritten as follows:

$$\begin{aligned}J_1(\mathbf{W}) &= \mathrm{tr}(((\mathbf{AW}^T\mathbf{X})^T(\mathbf{AW}^T\mathbf{X}) - \mathbf{I}_N)^2) \\ &= \mathrm{tr}(\mathbf{X}^T\mathbf{WA}^T\mathbf{AW}^T\mathbf{XX}^T\mathbf{WA}^T\mathbf{AW}^T\mathbf{X}) \\ &\quad - 2 \times \mathrm{tr}(\mathbf{X}^T\mathbf{WA}^T\mathbf{AW}^T\mathbf{X}) + N.\end{aligned} \tag{6}$$

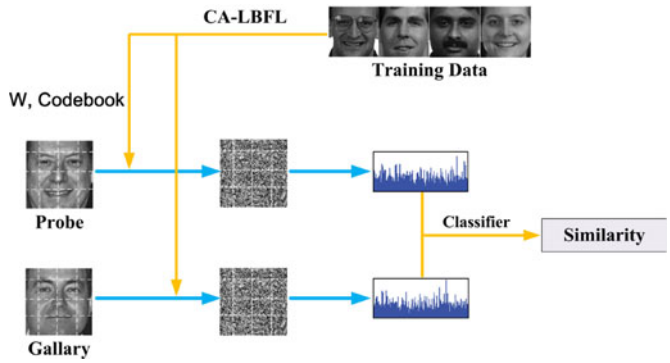


Fig. 5. The flow-chart of face representation approach based on CA-LBFL. We first divide each training face into several non-overlapped regions and learn the feature mapping **W** and the codebook for each region. Then, we apply the learned filter and the codebook to extract histogram feature for each block and concatenate them into a longer feature for face representation. Lastly, the similarity of face images is measured with the nearest neighbor classifier.

Similarly, we rewrite $J_3(\mathbf{W})$ and $J_4(\mathbf{W})$ as

$$\begin{aligned} J_3(\mathbf{W}) &= \|(\mathbf{W}^T\mathbf{X} - 0.5) \times \mathbf{1}^{N\times 1}\|_2^2 \\ &= \mathrm{tr}(\mathbf{W}^T\mathbf{X}\mathbf{1}^{N\times 1}\mathbf{1}^{1\times N}\mathbf{X}^T\mathbf{W}) \\ &\quad - N \times \mathrm{tr}(\mathbf{1}^{1\times K}\mathbf{W}^T\mathbf{X}\mathbf{1}^{N\times 1}) \\ &\quad + 0.25 \times \mathbf{1}^{1\times N}\mathbf{1}^{N\times 1} \end{aligned} \tag{7}$$

$$\begin{aligned} J_4(\mathbf{W}) &= \mathrm{tr}(\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}) - 2 \times \mathrm{tr}(\mathbf{W}^T\mathbf{X}\mathbf{M}^T\mathbf{W}) \\ &\quad + \mathrm{tr}(\mathbf{W}^T\mathbf{M}\mathbf{M}^T\mathbf{W}), \end{aligned} \tag{8}$$

where $\mathbf{M} \in \mathbb{R}^{d\times N}$ is the mean matrix which are repeated row vector of the mean of all PDVs.

While the objective function in (4) is not convex for **W** and **B** simultaneously, it is convex to one when fixing the other. Therefore, we optimize **W** and **B** using the following iterative approach.

Obtaining **B** *with a fixed W*: when **W** is fixed, the objective function in (4) can be rewritten as follows:

$$\min_{\mathbf{B}} J(\mathbf{B}) = ||(\mathbf{B} - 0.5) - \mathbf{W}^T\mathbf{X}||_F^2. \tag{9}$$

As **B** is a binary matrix, the solution can be directly obtained as

$$\mathbf{B} = 0.5 \times (\mathrm{sgn}(\mathbf{W}^T\mathbf{X}) + 1). \tag{10}$$

Learning **W** *with a fixed B*: when **B** is fixed, the objective function in (4) can be rewritten as follows:

$$\begin{aligned} \min_{\mathbf{W}} J(\mathbf{W}) &= \mathrm{tr}(\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}) \\ &\quad - 2 \times \mathrm{tr}(\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}) + \mathrm{tr}(\mathbf{W}^T\mathbf{Q}\mathbf{W}) \\ &\quad - 2 \times \lambda_1 \mathrm{tr}((\mathbf{B} - 0.5) \times \mathbf{X}^T\mathbf{W}) \\ &\quad - \lambda_2 \times N \times \mathrm{tr}(\mathbf{1}^{1\times K}\mathbf{W}^T\mathbf{X}\mathbf{1}^{N\times 1}) \\ \text{subject to} \quad & \mathbf{W}^T\mathbf{W} = \mathbf{I}, \end{aligned} \tag{11}$$

where

$$\begin{aligned} \mathbf{Q} \triangleq\ & \lambda_1\mathbf{X}\mathbf{X}^T + \lambda_2\mathbf{X}\mathbf{1}^{N\times 1}\mathbf{1}^{1\times N}\mathbf{X}^T \\ & - \lambda_3 \times (\mathbf{X}\mathbf{X}^T - 2\mathbf{X}\mathbf{M}^T + \mathbf{M}\mathbf{M}^T). \end{aligned} \tag{12}$$

We use gradient descent method with the curvilinear search algorithm to solve **W**. Algorithm 1 summarizes the detailed procedure of the proposed method.

3.2 Face Representation Using CA-LBFL

Having obtained the projection matrix **W**, we first project each PDV into a low-dimensional binary vector. Then, all binary codes within the same face region are represented as a histogram feature using a codebook learned from the training set by an unsupervised clustering method[1] for face representation. Lastly, features from all regions within a face are combined as the final representation of the whole face image. Fig. 5 illustrates the approach of face representation based on CA-LBFL.

Algorithm 1. CA-LBFL

Input: Training set $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$, iteration number T, parameters λ_1, λ_2 and λ_3, binary code length K, and convergence parameter ϵ
Output: Projection matrix **W**

```
Initialize W as the top K eigenvectors of XX^T corresponding to the K largest eigenvalues.
for t = 1, 2, ..., T do
    Update B with fixed W using (10).
    Update W with fixed B using (11).
    if |W^t − W^{t−1}| < ε and t > 2 then
        break
    end if
end for
return W
```

4 Context-Aware Local Binary Multi-Scale Feature Learning

While CA-LBFL learns discriminative features from image patches, the values of PDV in different scales are simply concatenated together as the input feature vector. However, each value in different scales has a specific characteristic, so that the naive operation of alignment loses the diversity of scales, which leads to a suboptimal result. In order to exploit the specific characteristic from different scales as well as the interactions of different projection matrices, we propose a context-aware local binary multi-scale feature learning method to jointly learn multiple projection matrices for mapping, where each projection matrix corresponds to a specific scale of PDV.

Suppose there are R vectors extracted from R different scales for each PDV, where each vector corresponds to a specific scale, and the length of each vector is l_r. Fig. 6 illustrates the approach to extract vectors in different scales at the same position. It is easy to prove that $l_r = 8r$. Let $\mathbf{x}_n^r \in \mathbb{R}^{l_r}$ be the vector in the rth scale of the nth PDV, and $\mathbf{X}_r = [\mathbf{x}_1^r, \mathbf{x}_2^r, \ldots, \mathbf{x}_N^r]$ be the N samples in r-scale. As aforementioned, R feature projection matrices need to be learned jointly for R scales, which separately project samples from different scales into context-aware binary codes $\mathbf{b}_n^r = [b_{1n}^r, b_{2n}^r, \ldots, b_{kn}^r]^T \in \{0,1\}^{K\times 1}$

1. In this work, the conventional K-means is used to learn the codebook due to its simplicity. While more sophisticated dictionary learning methods may further improve the performance of our approach, we don't consider them because it is out of the key scope of this work.

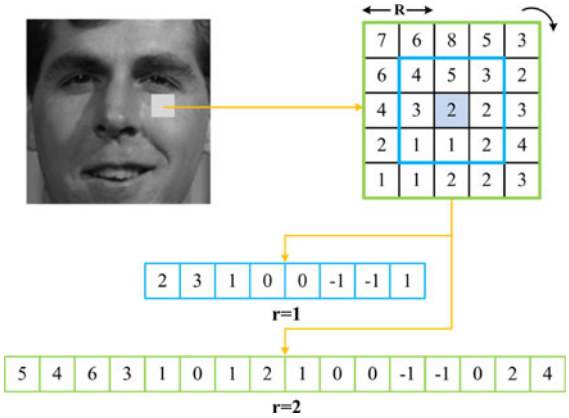


Fig. 6. An illustration to show how to extract PDV vectors in different scales at the same position. For pixels in each scale, we first compute the differences between the central pixel and these l_r neighboring pixels. Then, these differences are aligned as a vector, which becomes the input vector in this scale. R is selected as 2 in this figure for easy illustration, so that there are two vectors extracted with the lengths of 8 and 16, separately.

of the same length K. Similar to CA-LBFL, the learned binary codes should be context-aware, compact and distinctive. Also, the binary features extracted at the same position in different scales should be similar as possible, so that both discriminative and complementary information are exploited, simultaneously.

According to the above motivations, the following optimization objective function to context-aware multi-scale binary codes is formulated as follows:

$$\begin{aligned} \min_{\mathbf{W}_1,\ldots,\mathbf{W}_R,\boldsymbol{\alpha}} G &= \sum_{r=1}^{R} \alpha_r J(\mathbf{W}_r) + \lambda D(\mathbf{W}_1,\ldots,\mathbf{W}_R) \\ \text{subject to} \quad & \sum_{r=1}^{R} \alpha_r = 1,\ \alpha_r \geq 0, \end{aligned} \tag{13}$$

where

$$D(\mathbf{W}_1,\ldots,\mathbf{W}_R) = \sum_{n=1}^{N} \sum_{\substack{r_1,r_2=1 \\ r_1 \neq r_2}}^{R} ||\mathbf{b}_n^{r_1} - \mathbf{b}_n^{r_2}||^2. \tag{14}$$

$\mathbf{W}_r$ is the projection matrix in r-scale, λ is the parameter to balance the two terms and $\alpha = [\alpha_1,\ldots,\alpha_R]$ is the weighting vector.

The physical meaning of the first term in (13) is similar to CA-LBFL, which makes the learned binary codes context-aware, compact and distinctive. The objective of the second term in (13) is to minimize the gap between descriptors in different scales extracted at the same position.

Similar to CA-LBFL, we relax the sgn$(\cdot)$ function as its signed magnitude. Since it is not convex for learning $\mathbf{W}_1,\ldots,\mathbf{W}_R$, $\boldsymbol{\alpha}$ and $\mathbf{B}_1,\ldots,\mathbf{B}_R$ simultaneously, we use the following iterative approach to learn each of them with the others fixed, where $\mathbf{B}_r = 0.5 \times (\text{sgn}(\mathbf{W}_r^T\mathbf{X}_r) + 1) \in \{0,1\}^{K\times N}$ is the learned binary codes for all samples in r-scale.

Learning $\mathbf{W_r}$ *by fixing other parameters*: when $\mathbf{W}_1,\ldots,\mathbf{W}_{r-1},\mathbf{W}_{r+1},\ldots,\mathbf{W}_R$, $\boldsymbol{\alpha}$ and $\mathbf{B}_1,\ldots,\mathbf{B}_R$ are fixed, (13) can be rewritten as follows:

$$\min_{\mathbf{W}_r} G(\mathbf{W}_r) = \alpha_r J(\mathbf{W}_r) + \lambda D_r(\mathbf{W}_r), \tag{15}$$

where

$$D_r(\mathbf{W}_r) = \sum_{s=1,s\neq r}^{R} ||\mathbf{W}_s^T\mathbf{X}_s - \mathbf{W}_r^T\mathbf{X}_r||_F^2. \tag{16}$$

Therefore, (15) can be further re-written as

$$\begin{aligned} \min_{\mathbf{W}_r} G = & \ \text{tr}(\mathbf{X}_r^T\mathbf{W}_r\mathbf{A}^T\mathbf{A}\mathbf{W}_r^T\mathbf{X}_r\mathbf{X}_r^T\mathbf{W}_r\mathbf{A}^T\mathbf{A}\mathbf{W}_r^T\mathbf{X}_r) \\ & - 2\times\text{tr}(\mathbf{X}_r^T\mathbf{W}_r\mathbf{A}^T\mathbf{A}\mathbf{W}_r^T\mathbf{X}_r) + \text{tr}(\mathbf{W}_r^T\mathbf{Q}_r\mathbf{W}_r) \\ & - 2\times\lambda_1\text{tr}((\mathbf{B}_r - 0.5)\times\mathbf{X}_r^T\mathbf{W}_r) \\ & - \lambda_2\times N\times\text{tr}(\mathbf{1}^{1\times K}\mathbf{W}_r^T\mathbf{X}_r\mathbf{1}^{N\times 1}) \\ & - 2\times\lambda\sum_{s=1,s\neq r}^{R}\text{tr}(\mathbf{X}_r^T\mathbf{W}_r\mathbf{W}_s^T\mathbf{X}_s) \\ \text{subject to} \quad & \mathbf{W}_r^T\mathbf{W}_r = \mathbf{I}. \end{aligned} \tag{17}$$

where

$$\begin{aligned} \mathbf{Q}_r \triangleq & \ \lambda_1\mathbf{X}_r\mathbf{X}_r^T + \lambda_2\mathbf{X}_r\mathbf{1}^{N\times 1}\mathbf{1}^{1\times N}\mathbf{X}_r^T \\ & - \lambda_3\times(\mathbf{X}_r\mathbf{X}_r^T - 2\mathbf{X}_r\mathbf{M}_r^T + \mathbf{M}_r\mathbf{M}_r^T). \end{aligned} \tag{18}$$

Then, gradient descent method with the curvilinear search algorithm is used to obtain $\mathbf{W}_r$.

Obtaining $\mathbf{B_r}$ *by fixing other parameters*: similar to CA-LBFL, $\mathbf{B}_r$ is updated as follows when $\mathbf{B}_1,\ldots,\mathbf{B}_{r-1}$, $\mathbf{B}_{r+1},\ldots,\mathbf{B}_R$, $\boldsymbol{\alpha}$ and $\mathbf{W}_1,\ldots,\mathbf{W}_R$ are fixed

$$\mathbf{B}_r = 0.5\times(\text{sgn}(\mathbf{W}_r^T\mathbf{X}_r) + 1). \tag{19}$$

Learning $\boldsymbol{\alpha}$ *by fixing other parameters*: having obtained $\mathbf{W}_1,\ldots,\mathbf{W}_R$ and $\mathbf{B}_1,\ldots,\mathbf{B}_R$, we can update $\boldsymbol{\alpha}$ as follows:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} G(\boldsymbol{\alpha}) &= \sum_{r=1}^{R}\alpha_r J(\mathbf{W}_r) \\ \text{subject to} \quad & \sum_{r=1}^{R}\alpha_r = 1,\ \boldsymbol{\alpha}_r \geq 0. \end{aligned} \tag{20}$$

However, (20) leads to a solution that only the α_r corresponding to the minimum $J(\mathbf{W}_r)$ equals to one, and the others equal to zero. As this result simply exploits the best scale instead of the complementary information of multiple scales, we modify α_r into α_r^p, and the objective function can be rewritten as follows:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} G(\boldsymbol{\alpha}) &= \sum_{r=1}^{R}\alpha_r^p J(\mathbf{W}_r) \\ \text{subject to} \quad & \sum_{r=1}^{R}\alpha_r = 1,\ \boldsymbol{\alpha}_r \geq 0. \end{aligned} \tag{21}$$

In order to solve the optimization problem, Lagrange function is constructed

$$G(\boldsymbol{\alpha},\beta) = \sum_{r=1}^{R}\alpha_r^p J(\mathbf{W}_r) - \beta\left(\sum_{r=1}^{R}\alpha_r - 1\right). \tag{22}$$

Let $\frac{\partial G(\boldsymbol{\alpha},\beta)}{\partial \alpha_r}=0$, respectively, and $\frac{\partial G(\boldsymbol{\alpha},\beta)}{\partial \beta}=0$, we can obtain

$$p\alpha_1^{p-1}J(\mathbf{W}_1)=\cdots=p\alpha_R^{p-1}J(\mathbf{W}_R)=\beta \tag{23}$$

$$\sum_{r=1}^{R}\alpha_r-1=0. \tag{24}$$

Therefore, α_r can be updated as follows:

$$\alpha_r=\frac{(1/J(\mathbf{W}_r))^{1/(p-1)}}{\sum_{r=1}^{R}(1/J(\mathbf{W}_r))^{1/(p-1)}}. \tag{25}$$

Algorithm 2 summarizes the detailed procedure of the proposed CA-LBMFL method. Having obtained the projection matrices $\mathbf{W}_1,\ldots,\mathbf{W}_R$, the final binary feature can be represented by concatenating the R learned binary codes into a longer binary descriptor. The weighting vector $\boldsymbol{\alpha}$ is exploited when calculating the Hamming distance of different binary features.

Algorithm 2. CA-LBMFL

Input: Training set $\mathbf{X}_1,\ldots,\mathbf{X}_R$, iteration number T, parameters λ_1, λ_2, λ_3 and λ, binary code length K
Output: Projection matrices $\mathbf{W}_1,\ldots,\mathbf{W}_R$ and the weighting vector $\boldsymbol{\alpha}$

```
for r = 1, 2, ..., R do
    Initialize W_r as the top K eigenvectors of X_r X_r^T corresponding to the K largest eigenvalues.
    Initialize B_r = 0.5 × (sgn(W_r^T X_r) + 1).
    Initialize α_r = 1/R.
end for
for t = 1, 2, ..., T do
    for r = 1, 2, ..., R do
        Update W_r using (17).
        Update B_r using (19).
    end for
    Update α using (25).
end for
return W_1, ..., W_R and α
```

5 Coupled Context-Aware Local Binary Feature Learning

In this section, we propose coupled learning methods based on CA-LBFL and CA-LBMFL for heterogeneous face matching, respectively.

5.1 Coupled CA-LBFL

In recent years, heterogeneous face recognition has attracted much attention in recent years [47], [48], [49], [50]. Near infrared versus visible light and photo versus sketch are typical heterogeneous faces, which are captured under different environments or by different sensors. In this work, we propose a coupled CA-LBFL method for heterogeneous face recognition by minimizing the difference between heterogeneous faces at the feature level. Unlike CA-LBFL, C-CA-LBFL learns K pairs of hash functions to obtain context-aware local binary features of different modalities simultaneously, with the smallest gap between corresponding codes from heterogeneous face images. Fig. 7 illustrates how to learn pairs of hash functions to minimize the gap between corresponding binary codes.

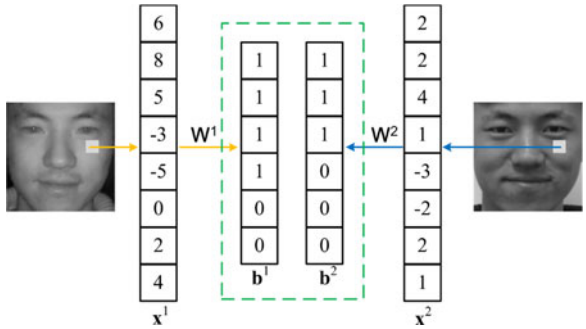


Fig. 7. An illustration to show how to learn pairs of hash functions to minimize the gap between corresponding binary codes from heterogeneous face images. For the two corresponding PDVs extracted from the same position of heterogeneous face images (NIR image for the left, and VIS image for the right), a pair of hash functions are learned to minimize the difference between the binary codes, as well as to follow the objective in CA-LBFL, respectively.

Let $\mathbf{X}^1=[\mathbf{x}_1^1,\mathbf{x}_2^1,\ldots,\mathbf{x}_N^1]$ and $\mathbf{X}^2=[\mathbf{x}_1^2,\mathbf{x}_2^2,\ldots,\mathbf{x}_N^2]$ be the N samples of the first and the second modality of the training heterogeneous face datasets, respectively, where $\mathbf{x}_n^1\in\mathbb{R}^d$ and $\mathbf{x}_n^2\in\mathbb{R}^d$ $(1\le n\le N)$ are corresponding PDVs extracted from the same position of a pair of heterogeneous face images. In C-CA-LBFL, we learn K pairs of hash functions to map and quantize $\mathbf{x}_n^1$ and $\mathbf{x}_n^2$ into context-aware binary vectors $\mathbf{b}_n^1=[b_{1n}^1,\ldots,b_{Kn}^1]^T\in\{0,1\}^{K\times 1}$ and $\mathbf{b}_n^2=[b_{1n}^2,\ldots,b_{Kn}^2]^T\in\{0,1\}^{K\times 1}$. Let $\mathbf{w}_k^1\in\mathbb{R}^d$ and $\mathbf{w}_k^2\in\mathbb{R}^d$ be the projection vectors for the kth function of each modality, respectively, and the kth binary codes b_{kn}^1 and b_{kn}^2 of $\mathbf{x}_n^1$ and $\mathbf{x}_n^2$ can be computed as

$$b_{kn}^1=0.5\times(\mathrm{sgn}((\mathbf{w}_k^1)^T\mathbf{x}_n^1)+1) \tag{26}$$

$$b_{kn}^2=0.5\times(\mathrm{sgn}((\mathbf{w}_k^2)^T\mathbf{x}_n^2)+1). \tag{27}$$

We formulate the following optimization objective function to make the learned binary codes context-aware as well as to minimize the gap between the codes of different modalities

$$\begin{aligned}\min_{\mathbf{w}_k^1,\mathbf{w}_k^2} J&=J_1+\lambda_1J_2+\lambda_2J_3+\lambda_3J_4+\lambda_4J_5\\&=\sum_{n=1}^{N}\sum_{m=1}^{2}||\sum_{k=1}^{K-1}||b_{kn}^m-b_{(k+1)n}^m||^2-1||^2\\&\quad+\lambda_1\sum_{n=1}^{N}\sum_{m=1}^{2}\sum_{k=1}^{K}\left\|(b_{kn}^m-0.5)-(\mathbf{w}_k^m)^T\mathbf{x}_n^m\right\|^2\\&\quad+\lambda_2\sum_{k=1}^{K}\sum_{m=1}^{2}\left\|\sum_{n=1}^{N}(b_{kn}^m-0.5)\right\|^2\\&\quad-\lambda_3\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{2}||b_{kn}^m-\mu_k^m||^2\\&\quad+\lambda_4\sum_{n=1}^{N}||b_n^1-b_n^2||^2,\end{aligned} \tag{28}$$

where the physical meanings of the first four terms in (28) are the same as those of CA-LBFL, which respectively ensure the learned binary codes context-aware, reduce the quantization loss, make each feature bit evenly distributed

and maximize the variance of the learned binary codes. The last term J_5 in (28) is to minimize the difference between the corresponding binary codes, so that the gap between the features learned from different modalities can be reduced.

Let $\mathbf{W}^1 = [\mathbf{w}_1^1, \mathbf{w}_2^1, \ldots, \mathbf{w}_K^1] \in \mathbb{R}^{d\times K}$ and $\mathbf{W}^2 = [\mathbf{w}_1^2, \mathbf{w}_2^2, \ldots, \mathbf{w}_K^2] \in \mathbb{R}^{d\times K}$ be the projection matrix of the first and the second modalities, so that each pair of samples $\mathbf{x}_n^1$ and $\mathbf{x}_n^2$ can be mapped into binary vectors as follows:

$$\mathbf{b}_n^1 = 0.5 \times (\mathrm{sgn}((\mathbf{W}^1)^T\mathbf{x}_n^1) + 1) \tag{29}$$

$$\mathbf{b}_n^2 = 0.5 \times (\mathrm{sgn}((\mathbf{W}^2)^T\mathbf{x}_n^2) + 1). \tag{30}$$

Then, (28) can be re-written into

$$\begin{aligned}\min_{\mathbf{W}^1,\mathbf{W}^2} J &= J_1 + \lambda_1 J_2 + \lambda_2 J_3 + \lambda_3 J_4 + \lambda_4 J_5 \\ &= \sum_{m=1}^{2} \mathrm{tr}((\mathbf{A}\mathbf{B}^m)^T(\mathbf{A}\mathbf{B}^m) - \mathbf{I}_N)^2 \\ &\quad + \lambda_1 \sum_{m=1}^{2} ||(\mathbf{B}^m - 0.5) - (\mathbf{W}^m)^T\mathbf{X}^m||_F^2 \\ &\quad + \lambda_2 \sum_{m=1}^{2} ||(\mathbf{B}^m - 0.5) \times \mathbf{1}^{N\times 1}||_F^2 \\ &\quad - \lambda_3 \sum_{m=1}^{2} \mathrm{tr}((\mathbf{B}^m - \mathbf{U}^m)^T(\mathbf{B}^m - \mathbf{U}^m)) \\ &\quad + \lambda_4 \mathrm{tr}((\mathbf{B}^1 - \mathbf{B}^2)^T(\mathbf{B}^1 - \mathbf{B}^2)),\end{aligned} \tag{31}$$

where $\mathbf{B}^1$ and $\mathbf{B}^2$ are the matrices of all binary codes and $\mathbf{U}^1$ and $\mathbf{U}^2$ are the mean matrices.

We also relax the sgn$(\cdot)$ function as its signed magnitude similar to CA-LBFL. Thus, (31) can be rewritten as follows:

$$\begin{aligned}\min_{\mathbf{W}} J(\mathbf{W}) &= \mathrm{tr}(\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}) \\ &\quad - 2 \times \mathrm{tr}(\mathbf{X}^T\mathbf{W}\mathbf{A}^T\mathbf{A}\mathbf{W}^T\mathbf{X}) + \mathrm{tr}(\mathbf{W}^T\mathbf{P}\mathbf{W}) \\ &\quad - 2 \times \lambda_1 \mathrm{tr}((\mathbf{B} - 0.5) \times \mathbf{X}^T\mathbf{W}) \\ &\quad - \lambda_2 \times 2N \times \mathrm{tr}(\mathbf{1}^{1\times 2K}\mathbf{W}^T\mathbf{X}\mathbf{1}^{2N\times 1}) \\ \text{subject to} \quad & \mathbf{W}^T\mathbf{W} = \mathbf{I}.\end{aligned} \tag{32}$$

where

$$\bar{\mathbf{X}} = \begin{bmatrix} \mathbf{X}^1 - \mathbf{M}^1 & 0 \\ 0 & \mathbf{X}^2 - \mathbf{M}^2 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{B}^1 & \mathbf{B}^2 \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}^1 \\ \mathbf{W}^2 \end{bmatrix}, \mathbf{X} = \begin{bmatrix} \mathbf{X}^1 & 0 \\ 0 & \mathbf{X}^2 \end{bmatrix}, \tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X}^1 \\ -\mathbf{X}^2 \end{bmatrix},$$

$$\begin{aligned}\mathbf{P} \triangleq\ & \lambda_1\mathbf{X}\mathbf{X}^T + \lambda_2\mathbf{X}\mathbf{1}^{2N\times 1}\mathbf{1}^{1\times 2N}\mathbf{X}^T \\ & - \lambda_3\bar{\mathbf{X}}\bar{\mathbf{X}}^T + \lambda_4\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T.\end{aligned} \tag{33}$$

We use the gradient descent method to solve the projection $\mathbf{W}$ similar to the Algorithm 1, and $\mathbf{W}$ is initialized as the top K eigenvectors of $(\lambda_1\mathbf{X}\mathbf{X}^T + \lambda_2\mathbf{X}\mathbf{1}^{2N\times 1}\mathbf{1}^{1\times 2N}\mathbf{X}^T - \lambda_3\bar{\mathbf{X}}\bar{\mathbf{X}}^T + \lambda_4\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T)$.

Having obtained $\mathbf{W}$, the codebook is learned respectively for each modalities on the heterogeneous face datasets. Similar to CA-LBFL, histogram features are extracted for each face region and then concatenated into a longer feature for face representation.

5.2 Coupled CA-LBMFL

Similar to coupled CA-LBFL, we also apply coupled learning method to CA-LBMFL for heterogeneous face recognition. The optimization objective function of coupled CA-LBMFL is as follows:

$$\begin{aligned}\min_{\mathbf{W}_r^i,\boldsymbol{\alpha}} G &= \sum_{r=1}^{R} \alpha_r J(\mathbf{W}_r^1, \mathbf{W}_r^2) + \lambda \sum_{i=1}^{2} D(\mathbf{W}_1^i, \ldots, \mathbf{W}_R^i) \\ \text{subject to} \quad & \sum_{r=1}^{R} \alpha_r = 1,\ \alpha_r \geq 0\end{aligned} \tag{34}$$

where

$$\begin{aligned}J(\mathbf{W}_r^1, \mathbf{W}_r^2) &= J_1 + \lambda_1 J_2 + \lambda_2 J_3 + \lambda_3 J_4 + \lambda_4 J_5 \\ &= \sum_{m=1}^{2} \mathrm{tr}(((\mathbf{A}\mathbf{B}_r^m)^T(\mathbf{A}\mathbf{B}_r^m) - \mathbf{I}_N)^2) \\ &\quad + \lambda_1 \sum_{m=1}^{2} ||(\mathbf{B}_r^m - 0.5) - (\mathbf{W}_r^m)^T\mathbf{X}_r^m||_F^2 \\ &\quad + \lambda_2 \sum_{m=1}^{2} ||(\mathbf{B}_r^m - 0.5) \times \mathbf{1}^{N\times 1}||_F^2 \\ &\quad - \lambda_3 \sum_{m=1}^{2} (\mathrm{tr}((\mathbf{B}_r^m - \mathbf{U}_r^m)^T(\mathbf{B}_r^m - \mathbf{U}_r^m)) \\ &\quad + \lambda_4 \mathrm{tr}((\mathbf{B}_r^1 - \mathbf{B}_r^2)^T(\mathbf{B}_r^1 - \mathbf{B}_r^2))\end{aligned} \tag{35}$$

$$D(\mathbf{W}_1^i, \ldots, \mathbf{W}_R^i) = \sum_{n=1}^{N} \sum_{\substack{r_1,r_2=1 \\ r_1\neq r_2}}^{R} ||\mathbf{b}_n^{r_1,i} - \mathbf{b}_n^{r_2,i}||^2. \tag{36}$$

The gradient decent method with the curvilinear search algorithm is used to obtain $\mathbf{W}_r^i$ and $\boldsymbol{\alpha}$, similar to the proposed methods in previous sections.

6 Experiments

We compared our methods with several state-of-the-art descriptors on three widely used homogeneous face datasets including LFW [51], YTF [52] and FERET [53], and a heterogeneous face database CASIA NIR-VIS 2.0 [54]. Then, we summarized the key observations of all the experiments. The followings describe the details of the experiments and results.

6.1 Homogeneous Face Recognition

In this section, we first evaluated the proposed CA-LBFL and CA-LBMFL methods on LFW, YTF and FERET databases. Then, we performed cross-dataset evaluation to test generalization ability of the proposed method and investigate the contributions of different terms to study the importance of the contextual information, respectively.

6.1.1 Results on LFW

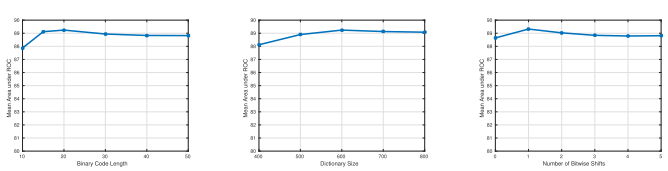
The LFW dataset [51] contains 13,233 face images of 5,749 subjects, which were captured from the web in wild conditions. Face images suffer from large intra-class variations such as varying poses, expressions, illuminations and backgrounds. In our experiments, we evaluated our CA-LBFL and CA-LBMFL with the unsupervised setting and the image-restricted with label-free outside data setting,

TABLE 1
Average Area Under ROC of LFW Dataset with the Unsupervised Setting versus Varying λ_1, λ_2, and λ_3

Parameters			AUC
$\lambda_1 = 10^2$	$\lambda_2 = 10^2$	$\lambda_3 = 10^7$	87.56
$\lambda_1 = 10^3$	$\lambda_2 = 10^2$	$\lambda_3 = 10^7$	88.32
$\lambda_1 = 10^3$	$\lambda_2 = 10^3$	$\lambda_3 = 10^7$	87.82
$\lambda_1 = 10^4$	$\lambda_2 = 10^2$	$\lambda_3 = 10^7$	87.94
$\lambda_1 = 10^4$	$\lambda_2 = 10^3$	$\lambda_3 = 10^7$	87.23
$\lambda_1 = 10^3$	$\lambda_2 = 10^2$	$\lambda_3 = 10^6$	87.54
$\lambda_1 = 10^3$	$\lambda_2 = 10^2$	$\lambda_3 = 10^8$	88.98
$\lambda_1 = 10^3$	$\lambda_2 = 10^2$	$\lambda_3 = 10^9$	**89.32**
$\lambda_1 = 10^3$	$\lambda_2 = 10^2$	$\lambda_3 = 10^{10}$	88.89
$\lambda_1 = 10^4$	$\lambda_2 = 10^3$	$\lambda_3 = 10^9$	89.02

respectively. We followed the standard evaluation protocol on the "View 2" dataset [51] including 3,000 matched pairs and 3,000 mismatched pairs, which were divided into 10 folds and each fold consisted of 300 matched (positive) pairs and 300 mismatched (negative) pairs. Assuming that the deviation was not too large, each face image was aligned with a conventional 2D affine transformation as preprocessing in our method, and then cropped into 128×128 to remove background information. In CA-LBFL and CA-LBMFL, each PDV was mapped into K-bit context-aware binary codes with the learned projection, and then encoded into histogram representation with the codebook. In our experiments, for CA-LBFL, different neighborhood radius sizes were examined by setting R as 2, 3 and 4, so that PDV was a 24-, 48-, and 80-dimensional vector for each pixel; for CA-LBMFL, R was fixed as 3, so that each PDV was divided into three vectors with the length of 8, 16, and 24, respectively. We applied the whitened PCA (WPCA) method to reduce the feature dimension into 500 to reduce the redundancy [9], [34]. For the unsupervised setting, the nearest neighbor classifier with the cosine similarity was used for face verification. For the image-restricted with label-free outside data setting, the discriminative deep metric learning (DDML) [55] method was used to learn discriminative similarity measure function for face verification.

Parameter Analysis. We first tested the mean area under ROC of the proposed CA-LBFL method with the unsupervised setting on "View1" dataset in LFW database with different parameters, and then applied these parameters for all following experiments including "View2" in LFW, YTF and FERET. We set R as 4, and examined the mean area under ROC versus different values of λ_1, λ_2 and λ_3 by fixing the binary code length K as 20 and the dictionary size as 600. The results are shown in Table 1, and three parameters λ_1, λ_2 and λ_3 were selected as 10^3, 10^2 and 10^9, respectively, to reach the best performance on LFW with the unsupervised setting.



(a) Code Length (b) Dictionary Size (c) Bitwise Shifts

Fig. 8. Average area under ROC (%) of LFW dataset with the unsupervised setting versus varying (a) binary code length, (b) dictionary size and (c) bitwise shifts.

TABLE 2
Mean Verification Rate (VR) (%) and Area Under ROC (AUC) (%) Comparison with State-of-the-Art Face Descriptors Under the Unsupervised Setting of the Standard LFW Protocol

Method	VR	AUC
LBP [1]	69.45	75.47
SIFT [16]	64.10	54.07
LARK [56]	72.23	78.30
POEM [9]	75.22	-
LHS [57]	73.40	81.07
MRF-MLBP [58]	80.08	89.94
PEM (LBP) [59]	81.10	-
PEM (SIFT) [59]	81.38	-
DFD [8]	84.02	-
High-dim LBP [60]	84.08	-
PAF [61]	-	94.05
CBFD [13]	-	88.65
CA-LBFL (R = 2)	81.50	86.44
CA-LBFL (R = 3)	82.97	88.92
CA-LBFL (R = 4)	83.30	89.24
CA-LBFL (R = 2 + 3 + 4)	84.72	91.66
CA-LBFL (combine)	**86.57**	**95.67**
CA-LBMFL (R = 3)	83.22	89.26

Then, binary code length K was tested when the dictionary size was fixed as 600. Fig. 8a shows that the best result was achieved when binary length was set to 20. Lastly, different dictionary sizes were examined and Fig. 8b shows that the dictionary size should be set as 600 to obtain the highest area under ROC. As local region was fixed to 8×8, each face image was represented as a 38,400-dimensional feature vector after using CA-LBFL ($38{,}400 = 500 \times 8 \times 8$).

For CA-LBMFL method, λ_1, λ_2, λ_3 and the dictionary size were selected the same as CA-LBFL, and λ was fixed to 10. The code length for each scale was set as 6, respectively, so that the total binary feature length K was 18.

In our algorithms, we set the preferred number of bitwise shifts as 1, and we also evaluated the mean area under ROC with different numbers of shifts. Fig. 8c shows that the best performance is achieved when 1 bit shift change is preferred, and the accuracy is decreased for other numbers bitwise changes. We conject the reason is that zero bitwise shift decreases the diversity of the learned binary codes, and more shifts weaken the constraint.

Comparison with the State-of-the-Art Methods. Table 2 tabulates the mean verification rate and area under ROC, and Fig. 9 shows the ROC curve of our CA-LBFL and CA-LBMFL compared with the state-of-the-art face descriptors with the unsupervised setting, respectively. We see that our CA-LBFL obtained better results than existing hand-crafted methods such as LARK and PEM, and achieved very competitive performance compared with existing learning-based methods such as DFD and CBFD. This is because our CA-LBFL learns context-aware local binary feature, which encodes more discriminative information and demonstrates stronger robustness to noise due to the contextual relationship. The performance of our CA-LBFL was further improved when multiple PDVs with different neighboring sizes were combined. PAF delivers an outstanding result on the unsupervised setting of LFW dataset. However, it

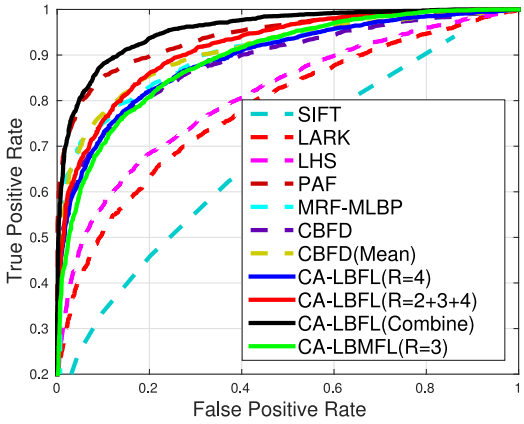


Fig. 9. ROC curves of different methods on LFW with the unsupervised setting.

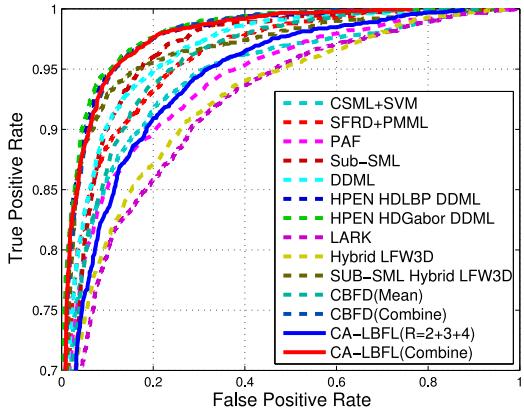


Fig. 10. ROC curves of different methods on LFW with the image-restricted with label-free outside data setting.

TABLE 3
Mean Verification Rate (VR) and the Standard Error (%) Comparison with State-of-the-Art Face Descriptors Under the Image-Restricted with Label-Free Outside Data Setting of the Standard LFW Protocol

Method	VR
LARK supervised [56]	85.10 ± 0.59
Convolutional DBN [30]	87.77 ± 0.62
STFRD+PMML [62]	89.35 ± 0.50
PAF [61]	87.77 ± 0.51
Sub-SML [63]	89.90 ± 0.38
VMRS [64]	91.10 ± 0.59
DDML [55]	90.68 ± 1.41
LM3L [65]	89.57 ± 1.53
HPEN+HD-LBP+DDML [66]	92.57 ± 0.36
HPEN+HD-Gabor+DDML [66]	$\mathbf{92.80 \pm 0.47}$
CBFD [13]	87.23 ± 1.68
CBFD (combine) [13]	92.62 ± 1.08
CA-LBFL (R = 2)	86.07 ± 1.37
CA-LBFL (R = 3)	87.34 ± 1.53
CA-LBFL (R = 4)	87.86 ± 1.41
CA-LBFL (R = 2 + 3 + 4)	89.21 ± 1.49
CA-LBFL (combine)	$\mathbf{92.75 \pm 1.13}$
CA-LBMFL (R = 3)	87.70 ± 1.22

TABLE 4
Comparison of Mean Verification Rate (VR) and the Standard Error (%) Under the Image-Restricted with Label-Free Outside Data Setting of the Standard LFW Protocol

Method	VR
Sparse SIFT+HOG+HDLBP	90.55 ± 1.44
CA-LBFL (combine)	$\mathbf{92.75 \pm 1.13}$

TABLE 5
Memory Cost of Each Local Descriptor (Bit), Final Feature Dimension and Computational Time (ms) of the Proposed CA-LBFL Method Compared with Different Feature Extraction Methods

Method	Memory Cost	Feature Dimension	Time
LBP [1]	16	3,776	22.9
SIFT [16]	1,024	8,192	63.7
DFD [8]	200	50,176	1,511.2
CA-LBFL	20	38,400	276.5

requires strong prior knowledge to design a pose-adaptive filter, and combines local Gabor filters for face representation. Our method is a general feature learning approach and does not require such prior knowledge. We evaluated our method with the same combination, and obtained 95.67 AUC (over 1.6 percent higher than PAF).

For the CA-LBMFL method, we see that it shows better performance than CA-LBFL with the same setting of $R = 3$, because the information of different scales are better exploited in CA-LBMFL.

Table 3 tabulates the average verification of the proposed CA-LBFL and CA-LBMFL methods, and Fig. 10 shows the ROC curves of our CA-LBFL as well as other state-of-the-art face descriptors for the image-restricted with label-free outside data setting, respectively. We see that our CA-LBFL achieved very competitive performance with the existing state-of-the-art methods, and outperformed most of them after combining three other existing hand-crafted descriptors including Sparse SIFT [55], HOG [55] and high-dimensional LBP [60]. Table 4 shows that the proposed CA-LBFL (combine) improves the mean verification rate by more than 2 percent compared with the combination of Sparse SIFT, HOG and HDLBP, which shows the effectiveness of the proposed approach. Also, CA-LBMFL performed better than CA-LBFL with the same R.

Computational Time. We compared the computational time with different face feature representation methods. Our hardware configuration comprises of a 2.8-GHz CPU and a 15G RAM. Table 5 shows the feature dimension and the computational time of our proposed method as well as other different methods. With higher feature dimensions, both DFD and CA-LBFL improved the recognition performance compared with LBP and SIFT. Moreover, our CA-LBFL is more efficient than DFD as only one PDV is extracted for one pixel, instead of a set of PDVs extracted in DFD.

As the optimization of $\mathbf{W}$ is one of the key steps in the proposed approach, we also evaluated the average optimization time of the proposed approach, and it took 86.27 s to train the projection matrix. The training time will not be largely affected by the number of samples, as all optimization steps are executed in matrix form.

TABLE 6
Recognition Accuracy and the Standard Error (%) Comparison with the Commonly Used Face Descriptors Under the Image-Restricted Setting of the Standard YTF Protocol

Method	Accuracy
LBP [1]	75.9 ± 1.4
SIFT [16]	76.4 ± 0.9
FPLBP [67]	73.6 ± 1.6
CSLBP [68]	73.7 ± 1.6
MBGS (LBP) [52]	76.4 ± 1.8
MBGS+SVM (LBP) [69]	78.9 ± 1.9
LE [35]	69.7 ± 2.1
DF.D [8]	78.1 ± 0.9
CA-LBFL (R = 2)	77.8 ± 0.9
CA-LBFL (R = 3)	79.1 ± 1.3
CA-LBFL (R = 4)	80.3 ± 1.0
CA-LBFL (R = 2 + 3 + 4)	$\mathbf{81.2 \pm 1.2}$
CA-LBMFL (R = 3)	79.4 ± 0.8

6.1.2 Results on YTF

The YouTube Face dataset [52] contains 3,425 videos from 1,595 different objects with varying variations of pose, expression and illumination, and the average length of each video clip is 181.3 frames. In our experiments, we followed the standard evaluation protocol [52] of unconstrained face verification including 5,000 video pairs, which were divided into 10 folds and each fold consists of 250 intra-personal pairs and 250 inter-personal pairs. We first learned feature representation using our CA-LBFL and CA-LBMFL for each frame of video clips, separately. As all face images have been aligned already, we averaged all descriptors of one video clip to make a mean vector as the feature of the video. Finally, we applied WPCA to reduce the feature dimension into 500. Similarly, DDML was used in the image-restricted setting for face verification.

Table 6 tabulates the average verification rate of our CA-LBFL, CA-LBMFL and the state-of-the-art learning-based face descriptors on YTF with the image-restricted setting. We see that our methods reached a higher verification rate than these commonly used descriptors. Moreover, the performance was further improved when multiple PDVs with different neighboring sizes were combined.

TABLE 7
Recognition Accuracy and the Standard Error (%) Comparison with the State-of-the-Art Face Verification Methods Under the Image-Restricted Setting of the Standard YTF Protocol

Method	Accuracy
APEM (LBP) [59]	77.4 ± 1.5
APEM (SIFT) [59]	78.5 ± 1.4
APEM (fusion) [59]	79.1 ± 1.5
STFRD+PMML [62]	79.5 ± 2.5
DDML (LBP) [55]	81.3 ± 1.6
DDML (combine) [55]	82.3 ± 1.5
EigenPEP [70]	84.8 ± 1.4
LM3L [55]	81.3 ± 1.2
CBFD [13]	78.2 ± 1.2
DeepFace [71]	$\mathbf{91.4 \pm 1.1}$
CA-LBFL (R = 2 + 3 + 4)	81.2 ± 1.2
CA-LBFL (combine)	$\mathbf{83.3 \pm 1.3}$

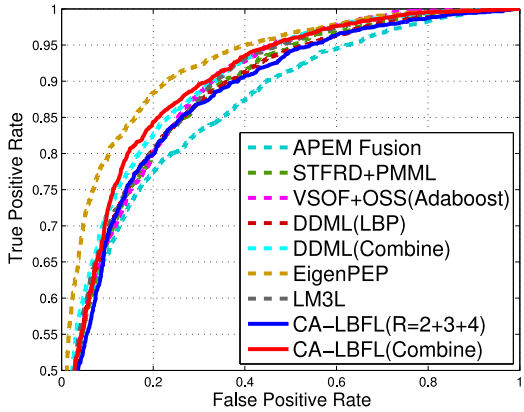


Fig. 11. ROC curves of different methods on YTF with the image-restricted setting.

Table 7 and Fig. 11 show the comparison between our method and the state-of-the-art face verification methods. While DeepFace [71] obtained the best results, our CA-LBFL still achieved very competitive performance with other state-of-the-art methods. Compared to DeepFace which requires large amounts of labeled samples for feature learning, our method is unsupervised and the number of parameters is heavily reduced. Better recognition rate is obtained after combining our CA-LBFL with LBP, CSLBP and FPLBP. Table 8 shows that there is a more than 3 percent improvement in accuracy for CA-LBFL (combine) compared with the combination of LBP, CSLBP and FPLBP.

6.1.3 Results on FERET

The FERET database is one of the largest publicly available databases, consisting of 13,539 face images of 1,565 subjects who are diverse across age, gender, and ethnicity. We followed the standard FERET evaluation protocol [53], where six sets including the *training*, *fa*, *fb*, *fc*, *dup1*, and *dup2* were constructed. According to the provided eye coordinates, all face images were aligned and cropped into 128×128 pixels. We performed feature learning on the *training* set and applied the learned projection on the other five sets for feature extraction. Lastly, we took *fa* as the gallery set, and the others as probe sets. For CA-LBFL, the codebook size was set to 500, and as the local region was fixed to 8×8, the dimension of feature vector after using CA-LBFL was 32,000. For CA-LBMFL, the codebook size was fixed to 600. Finally, we applied the whitened PCA (WPCA) method to project each sample into a 1,000-dimensional feature vector. Nearest neighbor classifier was used for face matching.

Table 9 tabulates the rank-one recognition rate of our methods as well as the state-of-the-art feature descriptors on FERET dataset. We see that our methods reached the best recognition rates on all four subsets, with the smallest gain of

TABLE 8
Comparison of Mean Accuracy and the Standard Error (%) Under the Image-Restricted Setting of the Standard YTF Protocol

Method	Accuracy
LBP + CSLBP + FPLBP	80.0 ± 1.4
CA-LBFL (combine)	$\mathbf{83.3 \pm 1.3}$

TABLE 9
Rank-One Recognition Rates (%) Comparison with State-of-the-Art Feature Descriptors with the Standard FERET Evaluation Protocol

Method	fb	fc	dup1	dup2
LBP [1]	93.0	51.0	61.0	50.0
LGBP [12]	94.0	97.0	68.0	53.0
LGT [6]	97.0	90.0	71.0	67.0
HGGP [11]	97.6	98.9	77.7	76.1
HOG [72]	90.0	74.0	54.0	46.6
LDP [10]	94.0	83.0	62.0	53.0
GV-LBP-TOP [7]	98.4	99.0	82.0	81.6
GV-LBP [7]	98.1	98.5	80.9	81.2
LQP [34]	**99.8**	94.3	85.5	78.6
POEM [9]	97.0	95.0	77.6	76.2
s-POEM [73]	99.4	**100.0**	91.7	90.2
DFD [8]	99.4	**100.0**	91.8	92.3
CBFD [13]	**99.8**	**100.0**	93.5	93.2
CA-LBFL (R = 2)	98.5	99.5	91.2	89.3
CA-LBFL (R = 3)	**99.8**	**100.0**	94.9	94.5
CA-LBFL (R = 4)	**99.8**	**100.0**	95.2	94.9
CA-LBMFL (R = 3)	**99.8**	**100.0**	**95.3**	**95.3**

1.8 percent in *dup1* set and 2.1 percent in *dup2* set. For CA-LBFL, compared with the state-of-the-art hand-crafted descriptors such as HGGP, GV-LBP-TOP and GV-LBP, CA-LBFL learns the feature descriptors in a data-driven way, which makes it more data-adaptive. As for recently proposed learning-based feature descriptors such as DFD and CBFD, our CA-LBFL is more stable and robust to noise due to the usage of contextual information. Therefore, higher recognition rates were obtained in our CA-LBFL method. For CA-LBMFL, besides all the advantages in CA-LBFL, it also exploits the specific characteristic from different scales, and delivered the best results in all four subsets.

6.2 Heterogeneous Face Recognition

In order to evaluate the effectiveness of the proposed method on heterogeneous faces, we further evaluated our C-CA-LBFL and C-CA-LBMFL on the CASIA NIR-VIS 2.0 dataset [54], which was used for heterogeneous face matching evaluation. The database consists of 725 subjects, with 1-22 VIS and 5-50 NIR face images per subject. Face images were cropped into 128 × 128 based on eye coordinates.

TABLE 10
Comparisons of the Rank-One Recognition Rate (%) and the Mean Verification Rate (%) with the Standard CASIA NIR-VIS 2.0 Evaluation Protocol, Where VR1 and VR2 Respectively Denote the Mean Verification Rate When the FAR Is Set to 0.1 and 1.0 Percent

Method	Rank1	VR1	VR2
CDFE [48]	27.9 ± 2.9	6.9	23.3
MvDA [47]	41.6 ± 4.1	19.2	42.8
GMLDA [74]	23.7 ± 1.4	5.1	16.6
GMMFA [74]	24.8 ± 1.1	7.6	19.5
LBP [1]	35.4 ± 2.7	4.2	31.8
LBP+LDA [1]	60.6 ± 2.4	23.9	52.7
TP-LBP [67]	36.2 ± 1.6	3.7	12.9
FP-LBP [67]	23.2 ± 1.0	1.7	9.0
SIFT [16]	49.1 ± 2.3	14.3	40.8
SIFT+LDA [16]	72.3 ± 1.5	35.9	63.1
C-CBFD [13]	56.6 ± 2.4	20.4	44.3
C-CBFD+LDA [13]	81.8 ± 2.3	47.3	75.3
C-CA-LBFL	58.3 ± 1.1	20.4	45.7
C-CA-LBFL+LDA	86.1 ± 0.9	51.6	80.8
C-CA-LBMFL	59.8 ± 0.5	21.2	45.4
C-CA-LBMFL+LDA	$\mathbf{87.9 \pm 0.9}$	**52.1**	**82.7**

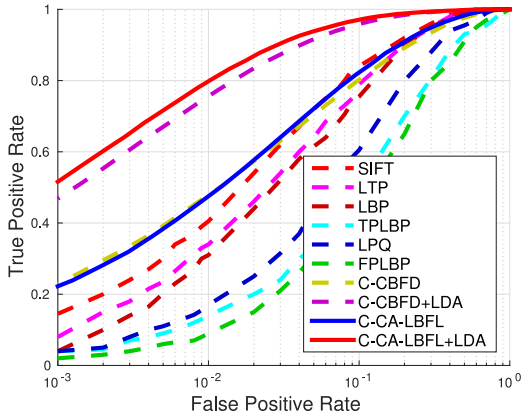


Fig. 12. ROC curves of different methods on CASIA NIR-VIS 2.0 dataset.

We followed the standard CASIA NIR-VIS 2.0 evaluation protocol, where the VIS images were taken as the gallery set and the NIR images as the probe set. First, DOG was used to pre-process each face image, which was used to learn the coupled filter as well as the codebook. Then, each histogram feature was projected into a 400-dimensional vector by WPCA. Lastly, the nearest neighbor classifier with the cosine similarity was utilized for face matching. Following the experiment on LFW dataset, parameters λ_1, λ_2 and λ_3 were set as 10^3, 10^2 and 10^9, respectively. λ_4 was selected as 10^3 and λ was select as 10 by cross validation. Table 10 tabulates the rank-one recognition rate and the mean verification rate under different FAR with the CASIA NIR-VIS 2.0 evaluation protocol, and Fig. 12 shows the ROC curves of different methods. In the experiments, we compared the proposed C-CA-LBFL with several commonly-used heterogeneous face recognition methods, such as common discriminant feature extraction (CDFE) [48], multi-view discriminant analysis (MvDA) [47], generalized multiview analysis (GMA) [74]. We see that our C-CA-LBFL outperformed most of other existing methods, and better performance was obtained after LDA is used for classification. Compared with the existing descriptor-based methods, our C-CA-LBFL learns a common subspace to reduce the modality difference, which is more effective to heterogeneous face matching. Compared with the existing learning-based methods which learn mappings to project heterogeneous faces into a common space, our C-CA-LBFL learns the modality-invariant descriptors at the feature level, which reduces the modality gap more effectively. The performance was further improved after a more sufficient utilization of difference scales for C-CA-LBMFL.

6.3 Comparison with CNN

We first conducted experiments on LFW and YTF to compare the proposed CA-LBFL and the state-of-the-art CNN methods by combining CA-LBFL with the pre-trained VGG-16 network, and Table 11 shows the experimental results. We observe that the proposed CA-LBFL+VGG

TABLE 11
Mean Verification Rate (VR) (%) and Number of Training Images Comparison with Different CNN Methods on the LFW and YTF Datasets

Method	LFW	YTF	Images
DeepFace [71]	97.4	91.4	4M
FaceNet [76]	98.9	**95.1**	200M
DeepID [33]	97.4	-	0.2M
DeepID2 [77]	99.1	-	0.2M
DeepID2+ [78]	99.4	93.2	0.3M
DeepID3 [79]	**99.5**	-	0.3M
VGG [80]	98.2	94.0	2.6M
CA-LBFL+VGG	**98.6**	**94.3**	2.6M

obtains competitive results with existing state-of-the-art CNN methods on both datasets and the combination increases the verification rate of VGG, which shows the effectiveness of the proposed approach.

Then, we compared our CA-LBFL with VGG network without outside face data by fine-tuning the pre-trained VGG-ImageNet [75] with different amount of facial images on the LFW dataset. On the one hand, it is very hard to train such a deep convolutional network from scratch with only thousands of facial images. On the other hand, ImageNet does not contain cropped and aligned faces, which maintains the fairness of the experiment. We flipped each facial image for data augmentation, and applied WPCA to reduce the feature dimension into 500 for all the methods. Table 12 shows the experimental results. We observe that VGG-ImageNet obtains 73.5 percent verification rate on the LFW dataset, which presents discriminativeness to faces to some extent. However, the performance improves slightly with only thousands of training data, because there are huge amount of parameters to train. Instead, the proposed CA-LBFL simply learns a projection for each local patch, which grasps the key properties of the learned binary codes. Therefore, deep face models still require millions of training samples, and our shallow models perform better on the small training set.

6.4 Evaluation on Other Applications

We conducted experiments on other applications to provide further demonstration of the effectiveness of the proposed methods. Table 13 shows the experimental results of CA-LBFL compared with three widely used local features including LBP [15], SIFT [16] and CBFD [13], where Brodatz [81], KTH-TIPS [82] and CUReT [83] are benchmark texture datasets, and Scene-15 [84] is a widely used scene dataset. We observe that the proposed CA-LBFL outperforms other compared local features on all the datasets.

TABLE 12
Comparison of Mean Verification Rate (VR) (%), Number of Training Facial Images and Training Manners Between the Fine-Tuned VGG-16 and CA-LBFL on the LFW Dataset

Method	VR	Facial Images	Training Manner
VGG	73.5	0	-
VGG	75.7	5K	Supervised
VGG	80.2	10K	Supervised
VGG	98.2	2.6M	Supervised
CA-LBFL	84.7	10K	Unsupervised

TABLE 13
Classification Results (%) of Different Methods on Various Texture and Scene Databases

Method	Brodatz	KTH-TIPS	CUReT	Scene-15
LBP	85.3	91.5	90.4	47.9
SIFT	81.9	86.3	84.6	72.2
CBFD	96.3	99.0	98.3	74.2
CA-LBFL	**98.6**	**99.4**	**99.2**	**76.7**

7 Conclusion

In this paper, we have proposed a context-aware local binary feature learning method for face recognition. In order to exploit more specific information from different scales, we have presented a context-aware local binary multi-scale feature learning method. Moreover, we have applied the above two methods to heterogeneous face matching by coupled learning methods (C-CA-LBFL and C-CA-LBMFL). Our methods achieve better or very competitive recognition performance on four widely used benchmark face databases compared with the state-of-the-art face descriptors. As our methods are general feature learning methods, it is reasonable and interesting to apply them to other computer vision applications such as object recognition and visual tracking in the future.

Acknowledgments

This work is supported by the National Key Research and Development Program of China under Grant 2016YFB 1001001, the National Natural Science Foundation of China under Grants 61672306, 61572271, 61527808, 61373074 and 61373090, the National 1000 Young Talents Plan Program, the National Basic Research Program of China under Grant 2014CB349304, the Ministry of Education of China under Grant 20120002110033, and the Tsinghua University Initiative Scientific Research Program.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces versus Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[3] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.

[4] J. Lu, Y.-P. Tan, and G. Wang, "Discriminative multimanifold analysis for face recognition from a single training sample per person," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 39–51, Jan. 2013.

[5] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[6] Z. Lei, S. Z. Li, R. Chu, and X. Zhu, "Face recognition with local Gabor textons," in *Proc. Int. Conf. Advances Biometrics*, 2007, pp. 49–57.

[7] Z. Lei, S. Liao, M. Pietikainen, and S. Z. Li, "Face recognition by exploring information jointly in space, scale and orientation," *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 247–256, Jan. 2011.

[8] Z. Lei, M. Pietikainen, and S. Z. Li, "Learning discriminant face descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 289–302, Feb. 2013.

[9] N.-S. Vu and A. Caplier, "Enhanced patterns of oriented edge magnitudes for face recognition and image matching," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1352–1365, Mar. 2012.

[10] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533–544, Feb. 2010.

[11] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of Gabor phase patterns (HGPP): A novel object representation approach for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 57–68, Jan. 2007.

[12] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, "Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 786–791.

[13] J. Lu, V. E. Liong, Z. Xiuzhuang, and J. Zhou, "Learning compact binary face descriptor for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2041–2056, Oct. 2015.

[14] J. Lu, V. E. Liong, and J. Zhou, "Simultaneous local binary feature learning and encoding for face recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3721–3729.

[15] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[17] H. Bay, T. Tuytelaars, and L. van Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[18] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2929–2936.

[19] A. Torralba, "Contextual priming for object detection," *Int. J. Comput. Vis.*, vol. 53, no. 2, pp. 169–191, 2003.

[20] J. Feng, B. Ni, D. Xu, and S. Yan, "Histogram contextualization," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 778–788, Feb. 2012.

[21] X. Gao, J. Zhong, J. Li, and C. Tian, "Face sketch synthesis algorithm based on E-HMM and selective ensemble," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 4, pp. 487–496, Apr. 2008.

[22] Z. Zhu, P. Luo, X. Wang, and X. Tang, "Deep learning identity-preserving face space," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 113–120.

[23] H. K. Galoogahi and T. Sim, "Inter-modality face sketch recognition," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2012, pp. 224–229.

[24] C. Peng, X. Gao, N. Wang, and J. Li, "Graphical representation for heterogeneous face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 2, pp. 301–312, Feb. 2017.

[25] D. Yi, R. Liu, R. Chu, Z. Lei, and S. Z. Li, "Face matching between near infrared and visible light images," in *Proc. Int. Conf. Advances Biometrics*, 2007, pp. 523–530.

[26] A. Mignon and F. Jurie, "CMML: A new metric learning approach for cross modal matching," in *Proc. Asian Conf. Comput. Vis.*, 2012, pp. 14–27.

[27] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2007, pp. 153–160.

[28] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 833–840.

[29] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[30] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2518–2525.

[31] A. Hyvärinen, J. Hurri, and P. O. Hoyer, "Independent component analysis," *Natural Image Statist.*, vol. 39, pp. 151–175, 2009.

[32] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng, "ICA with reconstruction cost for efficient overcomplete feature learning," in *Proc. Advances Neural Inf. Process. Syst.*, 2011, pp. 1017–1025.

[33] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1891–1898.

[34] S. U. Hussain, T. Napoléon, and F. Jurie, "Face recognition using local quantized patterns," in *Proc. British Mach. Vis. Conf.*, 2012, pp. 1–12.

[35] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2707–2714.

[36] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.

[37] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.

[38] S. Leutenegger, M. Chli, and R. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2548–2555.

[39] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 510–517.

[40] V. Balntas, L. Tang, and K. Mikolajczyk, "BOLD-binary online learned descriptor for efficient image matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2367–2375.

[41] B. Fan, Q. Kong, T. Trzcinski, Z. Wang, C. Pan, and P. Fua, "Receptive fields selection for binary feature description," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2583–2595, Jun. 2014.

[42] C. Strecha, A. M. Bronstein, M. M. Bronstein, and P. Fua, "LDAHash: Improved matching with smaller descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 66–78, Jan. 2012.

[43] T. Trzcinski and V. Lepetit, "Efficient discriminative projections for compact binary descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 228–242.

[44] T. Trzcinski, M. Christoudias, P. Fua, and V. Lepetit, "Boosting binary keypoint descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2874–2881.

[45] S. Zhang, Q. Tian, Q. Huang, W. Gao, and Y. Rui, "USB: Ultrashort binary descriptor for fast visual matching and retrieval," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3671–3683, Aug. 2014.

[46] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 817–824.

[47] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 808–821.

[48] D. Lin and X. Tang, "Inter-modality face recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 13–26.

[49] G. Lu, Y. Yan, L. Ren, P. Saponaro, N. Sebe, and C. Kambhamettu, "Where am I in the dark: Exploring active transfer learning on the use of indoor localization based on thermal imaging," *Neurocomputing*, vol. 173, pp. 83–92, 2016.

[50] J. Ma, C. Chen, C. Li, and J. Huang, "Infrared and visible image fusion via gradient transfer and total variation minimization," *Inf. Fusion*, vol. 31, pp. 100–109, 2016.

[51] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," UMass Amherst, Amherst, MA, USA, Tech. Rep. 07-49, 2007.

[52] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 529–534.

[53] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[54] S. Z. Li, D. Yi, Z. Lei, and S. Liao, "The CASIA NIR-VIS 2.0 face database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2013, pp. 348–353.

[55] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1875–1882.

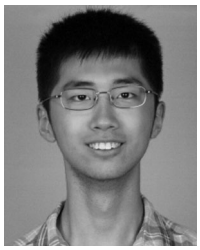
[56] H. J. Seo and P. Milanfar, "Face verification using the LARK representation," *IEEE Trans. Inf. Forensics Secur.*, vol. 6, no. 4, pp. 1275–1286, Dec. 2011.

[57] G. Sharma, S. ul Hussain, and F. Jurie, "Local higher-order statistics (LHS) for texture categorization and facial analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 1–12.

[58] S. R. Arashloo and J. Kittler, "Efficient processing of MRFs for unconstrained-pose face recognition," in *Proc. IEEE 6th Int. Conf. Biometrics: Theory Appl. Syst.*, 2013, pp. 1–8.

[59] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3499–3506.

[60] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3025–3032.
[61] D. Yi, Z. Lei, and S. Z. Li, "Towards pose robust face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3539–3545.
[62] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen, "Fusing robust face region descriptors via multiple metric learning for face recognition in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1–8.
[63] X. Cao, D. Wipf, F. Wen, G. Duan, and J. Sun, "A practical transfer learning algorithm for face verification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 3208–3215.
[64] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz, "Fast high dimensional vector multiplication face recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1960–1967.
[65] J. Hu, J. Lu, J. Yuan, and Y.-P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," in *Proc. Asian Conf. Comput. Vis.*, 2015, pp. 252–267.
[66] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 787–796.
[67] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2008, pp. 1–14.
[68] L. Wolf, T. Hassner, and Y. Taigman, "Similarity scores based on background samples," in *Proc. Asian Conf. Comput. Vis.*, 2010, pp. 88–97.
[69] L. Wolf and N. Levy, "The SVM-minus similarity score for video face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3523–3530.
[70] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt, "Eigen-PEP for video face recognition," in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 1–16.
[71] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1701–1708.
[72] E. Meyers and L. Wolf, "Using biologically inspired features for face processing," *Int. J. Comput. Vis.*, vol. 76, no. 1, pp. 93–104, 2008.
[73] N.-S. Vu, "Exploring patterns of gradient orientations and magnitudes for face recognition," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 2, pp. 295–304, Feb. 2013.
[74] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2160–2167.
[75] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Int. Conf. Learning Rep.*, pp. 1–14, 2015, http://www.robots.ox.ac.uk/~vgg/publications/2015/Simonyan15/
[76] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.
[77] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 1988–1996.
[78] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2892–2900.
[79] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: Face recognition with very deep neural networks," *arXiv preprint arXiv:1502.00873*, 2015, https://arxiv.org/abs/1502.00873
[80] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. British Mach. Vis. Conf.*, 2015, vol. 1, no. 3, pp. 1–12.
[81] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, vol. 66. New York, NY, USA: Dover, 1966.
[82] E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh, "On the significance of real-world conditions for material classification," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 253–266.
[83] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real-world surfaces," *ACM Trans. Graph.*, vol. 18, no. 1, pp. 1–34, 1999.
[84] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 2169–2178.



Yueqi Duan received the BE degree from the Department of Automation, Tsinghua University, Beijing, China, in 2010. He is currently working toward the PhD degree in the Department of Automation, Tsinghua University, Beijing, China. His research interests include visual recognition, feature learning, and binary descriptor.

Jiwen Lu received the BEng degree in mechanical engineering and the MEng degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, and the PhD degree in electrical engineering from the Nanyang Technological University, Singapore, respectively. He is currently an associate professor in the Department of Automation, Tsinghua University, China. His research interests include computer vision, pattern recognition, and machine learning, where he has authored/co-authored more than 150 scientific papers in these areas. He serves an associate editor of the *Pattern Recognition Letters*, the *Neurocomputing*, and the *IEEE Access*. He was a recipient of the Best Student Paper Award from Pattern Recognition and Machine Intelligence Association of Singapore in 2012, and the National 1000 Young Talents Plan Program in 2015, respectively. He is a senior member of the IEEE.

Jianjiang Feng received the BS and PhD degrees from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, China, in 2000 and 2007, respectively. He is an associate professor in the Department of Automation, Tsinghua University, Beijing. From 2008 to 2009, he was a post doctoral researcher in the PRIP Lab, Michigan State University. He is an associate editor of the *Image and Vision Computing*. His research interests include fingerprint recognition and computer vision. He is a member of the IEEE.

Jie Zhou received the BS and MS degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the PhD degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995. From then to 1997, he served as a postdoctoral fellow in the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a full professor in the Department of Automation, Tsinghua University. His research interests include computer vision, pattern recognition, and image processing. In recent years, he has authored more than 100 papers in peer-reviewed journals and conferences. Among them, more than 30 papers have been published in top journals and conferences such as the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, the *IEEE Transactions on Image Processing*, and CVPR. He is an associate editor of the *International Journal of Robotics and Automation* and two other journals. He received the National Outstanding Youth Foundation of China Award. He is a senior member of the IEEE.