Simultaneous Local Binary Feature Learning and Encoding for Face Recognition

Jiwen Lu[1], Venice Erin Liong[2], and Jie Zhou[1]
[1]Department of Automation, Tsinghua University, Beijing, China
[2]Rapid-Rich Object Search (ROSE) Lab, Interdisciplinary Graduate School,
Nanyang Technological University, Singapore
elujiwen@gmail.com; veniceer001@e.ntu.edu.sg; jzhou@tsinghua.edu.cn

Abstract

In this paper, we propose a simultaneous local binary feature learning and encoding (SLBFLE) method for face recognition. Different from existing hand-crafted face descriptors such as local binary pattern (LBP) and Gabor features which require strong prior knowledge, our SLBFLE is an unsupervised feature learning approach which is automatically learned from raw pixels. Unlike existing binary face descriptors such as the LBP and discriminant face descriptor (DFD) which use a two-stage feature extraction approach, our SLBFLE jointly learns binary codes for local face patches and the codebook for feature encoding so that discriminative information from raw pixels can be simultaneously learned with a one-stage procedure. Experimental results on four widely used face datasets including LFW, YouTube Face (YTF), FERET and PaSC clearly demonstrate the effectiveness of the proposed method.

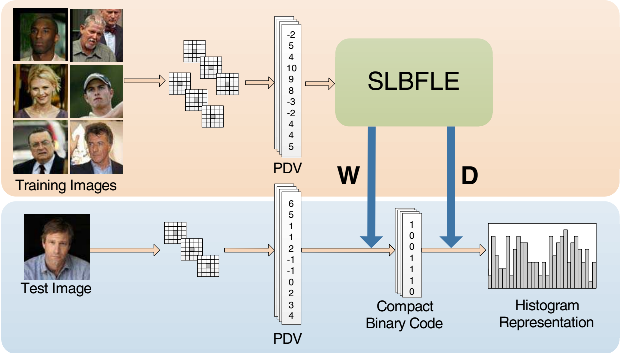


Figure 1. The basic idea of the proposed SLBFLE approach for face representation. For each training face image, we extract pixel difference vectors (PDVs) and jointly learn a discriminative mapping W and a dictionary D for feature extraction. The mapping is to project each PDV into a low-dimensional binary vector, and the dictionary is used as the codebook for feature local encoding. For each test image, the PDVs are first extracted and encoded into binary codes using the learned feature mapping, and then converted as a histogram feature with the learned dictionary.

1. Introduction

Face recognition is a classical and longstanding computer vision problem and a variety of face recognition algorithms have been proposed in the literature [1, 4, 23, 24, 38, 42, 52, 53, 54, 55]. Generally, there are two important procedures in a practical face recognition system: face representation and face matching. The aim of face representation is to extract discriminative feature descriptors to make face images more separable, and the objective of face matching is to design effective classifiers to differentiate different face patterns. In this work, we focus on the first one and present a new unsupervised feature learning approach for face representation.

Existing face representation methods can be mainly classified into two categories: holistic feature representation [4, 38] and local feature representation [1, 28]. Representative holistic feature representation methods include principal component analysis (PCA) [38] and linear discriminant analysis (LDA) [4], and typical local feature descriptors are local binary pattern (LBP) [1] and Gabor features [28]. While many face descriptors have been proposed in the literature [23, 24, 42, 52, 53, 54], most of them are hand-crafted and usually require strong prior knowledge to design. Moreover, some of them are computationally expensive, which may limit their practical applications.

Recently, feature learning has been successfully applied for face recognition. For example, Cao *et al.* [8] presented a learning-based (LE) feature representation method by applying the bag-of-word (BoW) framework. Hussain *et al.* [19] proposed a local quantized pattern (LQP) and Lei *et al.* [25] proposed a discriminant face descriptor (DFD) method to learn LBP-like features. Sun *et al.* [36] proposed a deep convolutional neural networks method to learn face representations. However, most of them learn real-valued face feature descriptors. For face recognition, binary features are more robust to local changes in face images because small variations caused by varying expressions and

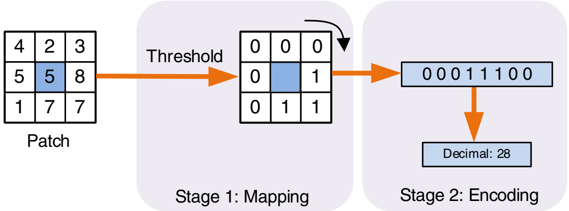


Figure 2. The basic idea of the LBP method, where a two-stage procedure is used for local feature extraction: feature mapping and feature encoding. For the feature mapping stage, the difference between the central pixel and the neighboring pixels are computed and binarized with a fixed threshold. For the feature encoding stage, the mapped binary codes are encoded as a real value by using a hand-crafted pattern coding strategy.

illuminations can be eliminated by quantized binary codes.

In this paper, we propose a new simultaneous local binary feature learning and encoding (SLBFLE) method for face recognition. Figure 1 illustrates the basic idea of our proposed approach. Motivated by the fact that binary features are robust to local changes such as varying illuminations and expressions [1, 20, 33, 34], we aim to learn compact binary codes directly from raw pixels for face representation. Unlike previous binary feature descriptors such as LBP and discriminant face descriptor (DFD) [25] which use a two-stage feature extraction approach, our proposed SLBFLE jointly learns binary codes for local face patches and the codebook for feature encoding so that discriminative information from raw pixels can be jointly learned with a one-stage procedure. Experimental results on four widely used face datasets including LFW, YouTube Face (YTF), FERET and PaSC clearly demonstrate the effectiveness of the proposed method.

2. Proposed Approach

In this section, we first review the LBP method, and then present the proposed SLBFLE method. Lastly, we show how to use SLBFLE for face representation.

2.1. Review of LBP

LBP is an effective feature descriptor in face recognition [1]. For each pixel in face image, LBP first computes the difference between the central pixel and the neighboring pixels and binarizes the difference with a fixed threshold. Secondly, these binary bins are encoded as a real value by using a hand-crafted pattern coding strategy. Figure 2 illustrates the basic idea of LBP, where two individual stages are used for feature representation.

There are two shortcomings in LBP: 1) both the binarization and feature encoding stages are hand-crafted, which are not optimal for local feature representation; 2) a two-stage procedure is used in LBP, which is not effective enough because some useful information for codebook learning may be compromised in the binarization stage. To address this, we propose a SLBFLE method to learn a discriminative mapping and a compact codebook for feature mapping and encoding jointly, so that more data-adaptive information can be exploited in the learned features. The following describes the details of the proposed method.

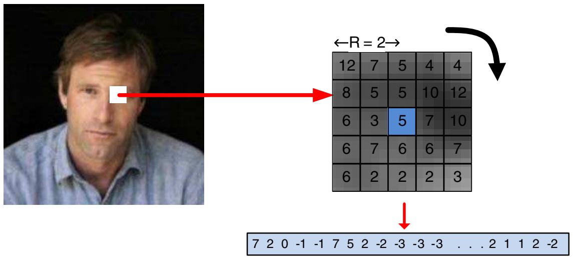


Figure 3. An illustration to show how to extract pixel difference vectors (PDV) from the original face image. Given a face patch whose size is $(2R+1)\times(2R+1)$, we first compute the difference between the central pixel and the neighboring pixels. Then, these differences are considered as a PDV. In this figure, R is selected as 2, so that there are 24 neighboring pixels selected and the PDV is a 24-dimensional feature vector.

2.2. SLBFLE

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{d\times N}$ be a set of N training samples, where $\mathbf{x}_n \in \mathbb{R}^d$ $(1 \leq n \leq N)$ is a pixel difference vector (PDV) extracted from an original face image. Figure 3 illustrates how to extract a PDV for a given face patch. Compared with the original raw pixel patch, PDV measures the difference between the central pixel and the neighboring pixels within a patch, so that it can better describe how pixel values change spatially and implicitly encode important visual patterns such as edges and lines in face images.

As aforementioned, our SLBFLE method aims to jointly learn a discriminative mapping and a dictionary for feature mapping and encoding. Assume there are K hash functions to be learned in SLBFLE, which map and quantize each $\mathbf{x}_n$ into a binary vector $\mathbf{b}_n = [\mathrm{b}_{n1}, \cdots, \mathrm{b}_{nK}] \in \{0, 1\}^{1\times K}$, so that the binary codes are learned automatically rather than using an empirical thresholding method. Let $\mathbf{w}_k \in \mathbb{R}^d$ be the projection vector for the kth function, the kth binary code b_{nk} of $\mathbf{x}_n$ can be computed as follows:

$$\mathrm{b}_{nk} = 0.5 \times (\mathrm{sgn}(\mathbf{w}_k^T \mathbf{x}_n) + 1) \tag{1}$$

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

Having obtained binary codes for these PDVs in the training set, we also require a codebook to pool those binary codes in each face image into a histogram feature. Previous methods applied the K-means algorithm to learn the codebook [8, 19, 25]. However, some useful information for codebook learning may be compromised in the mapping learning stage if they are learned sequentially. In this work, we learn them simultaneously so that discriminative information can be jointly exploited.

Let $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_C]$ and $\mathbf{A} = [\alpha_1, \alpha_2, \cdots, \alpha_N]$ be the dictionary and the corresponding representation coefficient matrix, respectively, where $\mathbf{d}_c \in \mathbb{R}^{1\times K}$ $(1 \le c \le C)$ is the cth atom in the dictionary, C is the total number of atoms in the dictionary, $\alpha_n \in \mathbb{R}^{1\times C}$ is the representation coefficient for $\mathbf{x}_n$. We formulate the following optimization problem:

$$\begin{aligned}
\min_{\mathbf{w},\mathbf{D},\alpha} J &= J_1 + \lambda_1 J_2 + \lambda_2 J_3 + \lambda_3 J_4 \\
&= \sum_{n=1}^{N} \left(\|(\mathrm{b}_n - 0.5) - \mathbf{D}\alpha_n\|^2 + \gamma\|\alpha_n\|_1 \right) \\
&+ \lambda_1 \sum_{n=1}^{N}\sum_{k=1}^{K} \|(\mathrm{b}_{nk} - 0.5) - \mathbf{w}_k^T \mathbf{x}_n\|^2 \\
&+ \lambda_2 \sum_{k=1}^{K} \| \sum_{n=1}^{N} (\mathrm{b}_{nk} - 0.5)\|^2 \\
&- \lambda_3 \|\mathbf{b}_{nk} - 0.5\|^2 \qquad (2)
\end{aligned}$$

where $\mathrm{b}_n = [\mathrm{b}_{n1}, \mathrm{b}_{n2}, \cdots, \mathrm{b}_{nK}]$ is the binary code vector for $\mathbf{x}_n$, and b_{nk} is the kth bit of b_n, λ_1, λ_2 and λ_3 are three parameters to balance the importance of different terms.

The objective of J_1 is to learn a dictionary D over the binary codes where each binary vector can be sparsely reconstructed by D. The goal of J_2 is to minimize the quantization loss between the original real-valued features and the binarized codes, so that most energy of the real-valued PDVs can be preserved in the learned binary codes. The physical meaning of J_3 is to ensure that each feature bit in the learned binary codes is evenly distributed over all the training samples (almost half of them are 1, and the other half are 0), so that the information conveyed by each bit is as large as possible. Finally, J_4 ensures that each projection vector results to independent and uncorrelated binary vectors.

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

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

The balancing constraint in J_3 can be relaxed by maximizing the variance for the kth bit as justified in [43]. Then, (2) can be re-written into the matrix form as follows:

$$\begin{aligned}
\min_{\mathbf{W},\mathbf{D},\mathbf{A}} J &= J_1 + \lambda_2 J_2 - \lambda_3 J_3 \\
&= \|(\mathbf{B} - 0.5) - \mathbf{D}\mathbf{A}\|_F^2 + \gamma\|\mathbf{A}\|_1 \\
&+ \lambda_1 \|(\mathbf{B} - 0.5) - \mathbf{W}^T\mathbf{X}\|_F^2 \\
&+ \lambda_2 \|(\mathbf{B} - 0.5) \times \mathbf{1}^{N\times 1}\|_F^2 \\
&- \lambda_3 \mathrm{tr}\left((\mathbf{B} - 0.5)(\mathbf{B} - 0.5)^T\right) \qquad (4)
\end{aligned}$$

where $B = 0.5 \times (\mathrm{sgn}(\mathbf{W}^T\mathbf{X}) + 1) \in \{0, 1\}^{K\times N}$ is the binary code matrix of all the training samples.

While the objective function in (4) is not convex for $\mathbf{D}$, $\mathbf{A}$, and $\mathbf{W}$, simultaneously, it is convex to one of them when the other two are fixed. We iteratively optimize $\mathbf{W}$, $\mathbf{D}$ and $\mathbf{A}$ by using the following iterative approach. We first initialize $\mathbf{W}$, $\mathbf{D}$ and $\mathbf{A}$ appropriate parameters and then iteratively update them sequentially as follows:

Step 1: Learning A with fixed W and D: when $\mathbf{W}$ and $\mathbf{D}$ are fixed, the objective function in (4) can be re-written as follows:

$$\min_{\mathbf{A}} J = \|(\mathbf{B} - 0.5) - \mathbf{D}\mathbf{A}\|_F^2 + \gamma\|\mathbf{A}\|_1 \qquad (5)$$

Since (5) is non-differentiable due to the sparsity function, standard unconstrained optimization techniques are infeasible and gradient-based methods cannot be applied directly. Instead, we optimize the objective function by decomposing it into a series of individual ℓ_1-regularized least square problem for α_n as follows:

$$\min_{\alpha_n} J = \sum_{n=1}^{N} (\|(\mathbf{b}_n - 0.5) - \mathbf{D}\alpha_n\|_2^2 + \gamma \sum_{j=1}^{K} |\alpha_n^{(j)}|) \qquad (6)$$

where α_n is the nth column of A, and $|\alpha_n^{(j)}|$ is the jth element of α_n. This optimization problem actually reflects a sparse coding problem which can already be solved by several optimization solutions [22, 49]. In this paper, we use the feature sign search algorithm in [22] to optimize α_n sequentially.

Step 2: Learning D with fixed W and A: when $\mathbf{W}$ and $\mathbf{A}$ are fixed, the optimization function in (4) can be re-written as the following objective function:

$$\begin{aligned}
\min_{\mathbf{D}} J &= \|(\mathbf{B} - 0.5) - \mathbf{D}\mathbf{A}\|_F^2 \\
\text{subject to:} \quad &\|\mathbf{d}_c\|^2 \le 1, 1 \le c \le C. \qquad (7)
\end{aligned}$$

The optimization objective function in (7) is a standard ℓ_2-constrained optimization problem. We use the conventional conjugate gradient decent method in [21] to optimize $\mathbf{D}$.

Step 3: Learning W with fixed D and A: when $\mathbf{D}$ and $\mathbf{A}$ are fixed, (4) can be re-written as follows:

$$\begin{aligned}
\min_{\mathbf{W}} J &= \|(\mathbf{B} - 0.5) - \mathbf{D}\mathbf{A}\|_F^2 \\
&+ \lambda_1 \|(\mathbf{B} - 0.5) - \mathbf{W}^T\mathbf{X}\|_F^2 \\
&- \lambda_2 \|(\mathbf{B} - 0.5) \times \mathbf{1}^{N\times 1}\|_F^2 \\
&+ \lambda_3 \mathrm{tr}\left((\mathbf{B} - 0.5)(\mathbf{B} - 0.5)^T\right) \qquad (8)
\end{aligned}$$

To our knowledge, (8) is an NP-hard problem due to the non-linear $\mathrm{sgn}(\cdot)$ function. To address this, we relax it with

```
Algorithm 1: SLBFLE
Input: Training set $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$, iteration
       number $T$, parameters $\lambda_1$, $\lambda_2$ and $\lambda_3$, and
       binary code length $K$.
Output: Projection $\mathbf{W}$, dictionary $\mathbf{D}$, and coefficient
        matrix $\mathbf{A}$.
Step 1 (Initialization):
1.1 Initialize $\mathbf{W}$ as the top $K$ eigenvectors of $\mathbf{X}\mathbf{X}^T$
corresponding to the $K$ smallest eigenvalues.
1.2 Initialize $\mathbf{D}$ and $\mathbf{A}$ with arbitrary initializations.
Step 2 (Optimization):
for $t = 1, 2, \cdots, T$ do
    Update $\mathbf{A}$ with fixed $\mathbf{W}$ and $\mathbf{D}$ using (5) .
    Update $\mathbf{D}$ with fixed $\mathbf{W}$ and $\mathbf{A}$ using (7) .
    Update $\mathbf{W}$ with fixed $\mathbf{D}$ and $\mathbf{A}$ using (9) .
    If $|\mathbf{W}^t - \mathbf{W}^{t-1}| < \epsilon$ and $t > 2$, go to Step 3.
end
Step 3 (Output):
Output the projection matrix $\mathbf{W}$, dictionary $\mathbf{D}$, and
coefficient matrix $\mathbf{A}$.
```

the signed magnitude [12, 43] and rewrite it as follows:

$$\begin{aligned} \min_{\mathbf{W}} J &= \|\mathbf{W}^T\mathbf{X} - 0.5 - \mathbf{D}\mathbf{A}\|_F^2 \\ &+ \lambda_1\|(\mathbf{B} - 0.5) - \mathbf{W}^T\mathbf{X}\|_F^2 \qquad (9) \\ &- \lambda_2 \mathrm{tr}(\mathbf{1}^{1\times K}\mathbf{W}^T\mathbf{X}\mathbf{1}^{N\times 1}) \\ &- \lambda_3 \mathrm{tr}(\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}) \\ &= (1 + \lambda_1 - \lambda_3)\mathrm{tr}(\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}) \\ &- 2\lambda_1 \mathrm{tr}((\mathbf{B}^T - 0.5)\mathbf{W}^T\mathbf{X}) \\ &- 2\mathrm{tr}((\mathbf{W}^T\mathbf{X})^T\mathbf{D}\mathbf{A}) \\ &- \lambda_2 \mathrm{tr}(\mathbf{1}^{1\times K}\mathbf{W}^T\mathbf{X}\mathbf{1}^{N\times 1}) \qquad (10) \end{aligned}$$

We use the gradient descent method with the curvilinear search algorithm in [44] to solve $\mathbf{W}$.

We repeat the above three steps until the algorithm is convergent. **Algorithm 1** summarized the detailed procedure of the proposed SLBFLE method.

2.3. SLBFLE-based Face Representation

Having obtained the feature mapping $\mathbf{W}$ and the dictionary $\mathbf{D}$, we first project each PDV into a low-dimensional binary vector and encode it as a real value. Then, all PDVs within the same face region is represented as a histogram feature. Finally, these features from all blocks within a face are concatenated as the feature representation of the whole face image. Figure 4 illustrates how to use the proposed SLBFLE for face representation.

3. Experiments

We conducted face recognition experiments on four widely used face datasets including LFW, YTF, FERET and PaSC. The followings describe the details of the experiments and results.

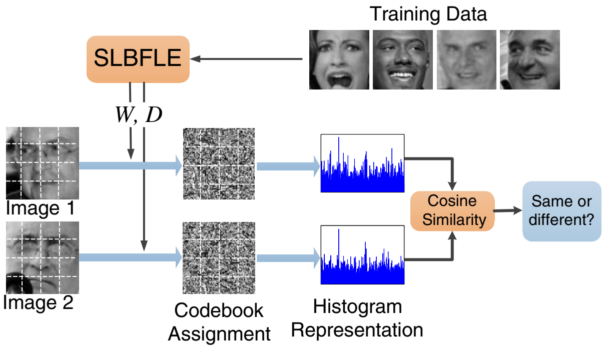
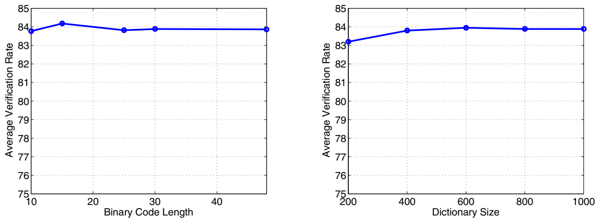


Figure 4. The flow-chart of the SLBFLE-based face representation approach. For each training face, we first divide it into several non-overlapped regions and learn the feature mapping $\mathbf{W}$ and dictionary $\mathbf{D}$ for each region. Then, we applied the learned filter and dictionary to extract histogram feature for each block and concatenated into a longer feature vector for face representation. Finally, the cosine similarity measure is used to measure face similarity for verification.

3.1. Results on LFW

The LFW dataset [17] contains 13233 images from 5749 persons. Facial images in this dataset were collected from the web, so that there are large intra-class variations in pose, illumination and expression because these images are captured in wild conditions. In our experiments, we evaluated our proposed method with the unsupervised setting and the image-restricted with label-free outside data setting. We followed the standard evaluation protocol on the "View 2" dataset [17] which includes 3000 matched pairs and 3000 mismatched pairs and is divided into 10 folds, where each fold consists of 300 matched (positive) pairs and 300 mismatched (negative) pairs. We used the aligned LFW-a dataset[1] for our evaluation, where each face image in LFW was aligned and cropped into 128×128 to remove the background information. We learned feature representation with our proposed SLBFLE. Specifically, each PDV was first projected into a K-bit binary codes with the learned projection $\mathbf{W}$ and then encoded as a feature with the learned dictionary $\mathbf{D}$. The parameters λ_1, λ_2, and λ_3 were empirically tuned as 0.001, 0.001 and 0.01, respectively, by using a cross-validation strategy on the "View 1" subset of the LFW dataset. We tested our method with different neighborhood radius sizes (R is set as 2, 3 and 4), which yields a 24-, 48-, and 80-dimensional PDV, respectively. We further applied the whitened PCA (WPCA) method to project each sample into a 500-dimensional feature vector to reduce the redundancy. For the unsupervised setting, the nearest neighbor classifier with the cosine similarity was used for

[1] Available: http://www.openu.ac.il/home/hassner/data/lfwa/.



(a) Binary Length (b) Dictionary Size

Figure 5. Mean verification rate of our method versus varying (a) binary code length and (b) dictionary size on LFW with the unsupervised setting.

Table 1. Mean verification rate of our SLBFLE versus different block sizes on LFW with the unsupervised setting.

Block size	4×4	6×6	8×8	10×10	12×12
Accuracy	81.43	83.20	**84.18**	83.87	82.77

face verification. For the image-restricted with label-free outside data setting, we used the discriminative deep metric learning (DDML) [15] method to learn discriminative similarity measure function for face verification.

Parameter Determination: We first tuned the parameters of our method with the unsupervised setting on LFW and applied these parameters for all the following experiments. We first set the dictionary size C as 600 and examined the performance of our proposed method versus different binary code length on LFW. Figure 5(a) shows the mean verification rate of our method versus different binary code length. We see that the best verification rate can be obtained when the length is set as 15.

Figure 5(b) shows the mean verification rate of our method versus different dictionary sizes. We find that our method achieves the best verification performance when the dictionary size is set as 600.

Comparison with the State-of-the-Art Methods: Table 2 tabulates the average verification rate and Figure 6 shows the ROC curve of our SLBFLE on LFW with the unsupervised setting, as well as those of the state-of-the-art face feature descriptors. We see that SLBFLE achieves better performance than existing hand-crafted feature descriptors such as LARK and PEM, and obtains very competitive performance with the existing learning-based feature descriptors such as DFD[2]. Moreover, the performance of SLBFLE can be further improved when multiple PDVs with different neighboring sizes are combined.

Table 3 tabulates the average verification rate and Figure 7 shows the ROC curve of our SLBFLE on LFW with the image-restricted with label-free outside data setting, as well as those of the state-of-the-art face verification methods. We see that our SLBFLE method with multiple PDVs extracted from different neighboring sizes outperforms most of the current state-of-the-art methods. Moreover, the performance of SLBFLE can be further boosted when it is combined with several other existing hand-crafted feature descriptors[3].

Table 2. Mean verification rate (VR) (%) and area under ROC (AUC) comparison with state-of-the-art face descriptors on LFW with the unsupervised setting.

Method	VR	AUC
LBP [39]	69.45	75.47
SIFT [39]	64.10	54.07
LARK [32]	72.23	78.30
POEM [41]	75.22	-
LHS [35]	73.40	81.07
MRF-MLBP [2]	80.08	89.94
PEM (LBP) [26]	81.10	-
PEM (SIFT) [26]	81.38	-
DFD [25]	84.02	-
High-dim LBP [9]	84.08	-
PAF [50]	**87.77**	**94.05**
SLBFLE (R=2)	82.02	88.95
SLBFLE (R=3)	84.08	90.46
SLBFLE (R=4)	84.18	90.53
SLBFLE (R=2+3+4)	**85.62**	**92.00**

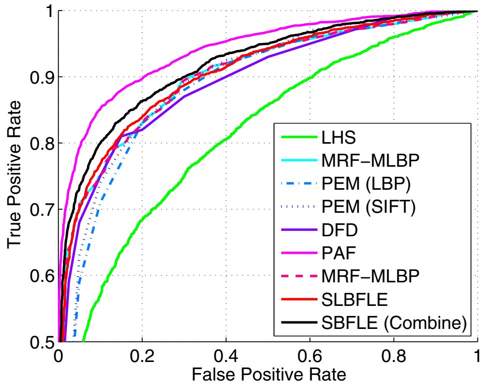


Figure 6. ROC curve of different face descriptors on LFW with the unsupervised setting.

3.2. Results on YTF

The YTF dataset [45] contains 3425 videos of 1595 different persons collected from the YouTube website. There are large variations in pose, illumination, and expression in each video, and the average length of each video clip is 181.3 frames. In our experiments, we followed the stan-

[2] Compared with DFD which is a supervised feature learning approach, our SLBFLE is unsupervised so that it is more convenient for practical applications.

[3] We combined our SLBFLE with 5 other existing feature descriptors including the Sparse SIFT [15], Dense SIFT [15], low-dimensional LBP [15], HOG [15], and high-dimensional LBP [9], the mean verification rate can be further improved by 2.79%, which outperforms the current best method by 0.17%.

Table 3. Mean verification rate and the standard error (%) comparison with state-of-the-art face verification methods on LFW with the image-restricted with label-free outside data setting.

Method	Accuracy
CSML+SVM [30]	88.00 ± 0.37
High-Throughput BIF [10]	88.13 ± 0.58
LARK supervised [32]	85.10 ± 0.59
DML-eig combined [51]	85.65 ± 0.56
Covolutional DBN [18]	87.77 ± 0.62
STFRD+PMML [11]	89.35 ± 0.50
PAF [50]	87.77 ± 0.51
Sub-SML [7]	89.90 ± 0.38
VMRS [3]	91.10 ± 0.59
DDML [15]	90.68 ± 1.41
LM3L [16]	89.57 ± 0.02
Hybrid on LFW3D [13]	85.63 ± 0.005
Sub-SML + Hybrid on LFW3D [13]	91.65 ± 0.01
HPEN + HD-LBP + DDML [56]	92.57 ± 0.003
HPEN + HD-Gabo + DDML [56]	92.80 ± 0.005
SLBFLE (R=2)	85.62 ± 1.41
SLBFLE (R=3)	86.57 ± 1.65
SLBFLE (R=4)	87.45 ± 1.28
SLBFLE (R=2+3+4)	90.18 ± 1.89
SLBFLE (All combined)	**92.97 ± 1.20**

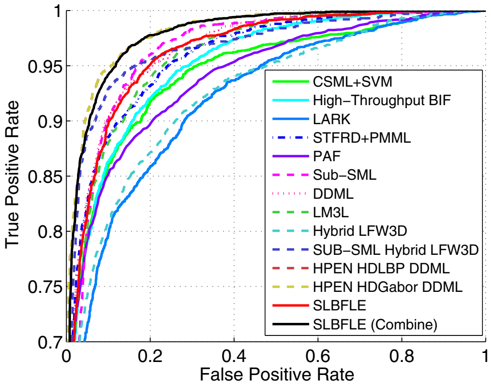


Figure 7. ROC curve of different face verification methods on LFW with the image-restricted with label-free outside data setting.

dard evaluation protocol [45] and tested our method for unconstrained face verification with 5000 video pairs. These pairs are equally divided into 10 folds, and each fold has 250 intra-personal pairs and 250 inter-personal pairs. For each video clip, we first learned feature representation for each frame by using our SLBFLE method, and then averaged all the feature vectors within one video clip to form a mean feature vector in our experiments because all face images have been aligned by the detected facial landmarks. Lastly, we used WPCA to project each mean vector into a 500-dimensional feature vector. Similarly, we also used the DDML method for face verification with the image-restricted setting.

Table 4. Comparisons of the mean verification rate and standard error (%) with state of the art learning-based face descriptors on YTF under the image-restricted setting.

Method	Accuracy
LBP [1]	75.86 ± 1.42
FPLBP [46]	73.58 ± 1.62
CSLBP [47]	73.70 ± 1.63
LE [8]	69.72 ± 2.06
DFD [25]	78.10 ± 0.94
SLBFLE (R=2)	80.35 ± 0.84
SLBFLE (R=3)	81.24 ± 1.32
SLBFLE (R=4)	82.36 ± 1.01
SLBFLE (R=2+3+4)	**82.88 ± 1.01**

Table 5. Comparisons of the mean verification rate and standard error (%) with the state-of-the-art face verification methods on YTF under the image-restricted setting.

Method	Accuracy
MBGS(LBP) [45]	76.4 ± 1.8
MBGS+SVM (LBP) [48]	78.9 ± 1.9
APEM(fusion) [26]	79.1 ± 1.5
STFRD+PMML [11]	79.5 ± 2.5
VSOF+OSS [14]	79.7 ± 1.8
DDML (LBP) [15]	81.3 ± 1.6
DDML (combined) [15]	82.3 ± 1.5
EigenPEP [27]	84.8 ± 1.4
LM3L [15]	81.3 ± 1.2
DeepFace [37]	**91.4 ± 1.1**
SLBFLE (R = 2+3+4)	82.9 ± 1.0
SLBFLE (R= 2+3+4 + LBP + CSLBP+ FPLBP)	**83.4 ± 1.0**

Table 4 tabulates the average verification rate of our method and three state-of-the-art learning-based face descriptors on YTF. We see that our method outperforms these state-of-the-art methods with the smallest gain of 2.25% in terms of the mean verification rate. Moreover, the performance of SLBFLE can be further improved when multiple PDVs with different neighboring sizes are combined.

Table 5 tabulates the average verification rates and Figure 8 shows the ROC curves of our method and state-of-the-art face verification methods on YTF. We see that our method achieves very competitive performance with state-of-the-art methods. DeepFace [37] delivers the best result for the YTF dataset however by combining all 6 features we are able to achieve competitive results with the other compared methods such as Eigen-Pep [27] and DDML [15].

3.3. Results on FERET

The FERET dataset consists of 13539 face images of 1565 subjects who are diverse across age, gender, and ethnicity. We followed the standard FERET evaluation protocol [31], where six sets including the *training*, *fa*, *fb*, *fc*, *dup1*, and *dup2* were constructed for experiment, respectively. All face images were scaled and cropped into

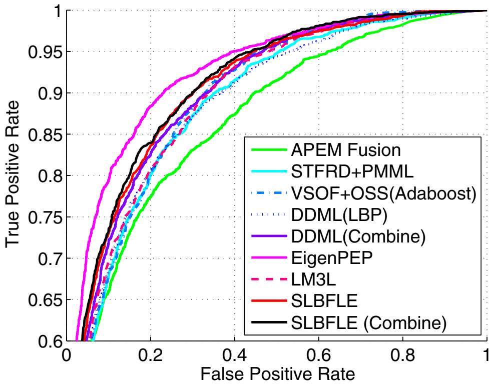


Figure 8. ROC curve of different face verification methods on YTF under the image-restricted setting.

128 × 128 pixels according to the provided eye coordinates. We performed feature learning on the generic *training* set, and applied the learned projection and dictionary matrix on the other five sets for feature extraction. Finally, we take *fa* as the gallery set and the other four sets as the probe sets. We followed the same parameter settings which were tuned on LFW. We applied WPCA to project each sample into a 1196-dimensional feature vector and applied the nearest neighbor classifier with the cosine similarity for face identification.

Table 6 tabulates the rank-one identification rate of our method, as well as the state-of-the-art feature descriptors on the FERET dataset. We see that our SLBFLE achieves the best recognition rate on all four subsets. Specifically, SLBFLE achieves much better performance than hand-crafted feature descriptors such as HGGP, GV-LBP-TOP and GV-LBP. This is because our SLBFLE is a data-adaptive feature representation method. Compared with the recently proposed learning-based feature representation methods such as DFD, our SLBFLE is a binary code based feature descriptor which can demonstrate stronger robustness to local variations. Hence, higher recognition rates can be obtained.

3.4. Results on PaSC

The PaSC dataset [5, 6] compose of 9376 images from 293 people which is separated to a query and target set having 4688 still images each. These face images were captured in different viewpoints, pose and distance from the camera making it a difficult face recognition dataset. Each still image is aligned using the provided eye coordinates and cropped in an image size of 128 × 128. We performed feature learning on a separate training set provided by PaSC and then implemented feature extraction for the query and target set. The extracted features are then projected using WPCA and reduce it into a 500-dimensional face representation. Following the standard evaluation protocol in [6], we compare the images in the query set to the target set and obtain a similarity matrix. We compare our face representation method with the LRPCA baseline provided in [6], two hand crafted descriptors, LBP and SIFT, and one feature learning method, DFD. The verification rates are tabulated in Table 7 and the ROC curve is shown in Fig. 9. As can seen, our proposed SLBFLE outperforms the handcrafted features, where the minimal improvement of verification rate is 4.2%. It is also comparable to a feature learning technique, DFD, with only a difference of 1.4%.

Table 6. Rank-one recognition rates (%) comparison with state-of-the-art feature descriptors with the standard FERET evaluation protocol.

Method	fb	fc	dup1	dup2
LBP [20]	93.0	51.0	61.0	50.0
LGBP [54]	94.0	97.0	68.0	53.0
HGGP [53]	97.6	98.9	77.7	76.1
LDP [52]	94.0	83.0	62.0	53.0
GV-LBP-TOP [24]	98.4	99.0	82.0	81.6
GV-LBP [24]	98.1	98.5	80.9	81.2
LQP [19]	99.8	94.3	85.5	78.6
POEM [42]	97.0	95.0	77.6	76.2
s-POEM [40]	99.4	**100.0**	91.7	90.2
DFD [25]	99.4	**100.0**	91.8	92.3
SLBFLE (R=2)	99.7	99.7	89.9	80.0
SLBFLE (R=3)	**99.9**	**100.0**	94.5	90.9
SLBFLE (R=4)	**99.9**	**100.0**	**95.2**	**92.7**

Table 7. Verification rate (%) at the 1.0% FAR of different methods on the PaSC dataset.

Method	Verification rate
LRPCA [6]	10.0
LBP [20]	25.1
SIFT [29]	23.2
DFD [25]	**30.6**
SLBFLE (R=2)	20.2
SLBFLE (R=3)	26.3
SLBFLE (R=4)	29.2

3.5. Analysis

Cross-Dataset Evaluation: To further evaluate the efficiency of our SLBFLE method, we perform cross-dataset experiment in which we learn our SLBFLE features using a different training set. In this experiment, we use the FERET training set to learn the parameters for feature extraction, $\mathbf{W}$ and $\mathbf{D}$, and evaluate it in the LFW dataset. After which we use the View 1 training set of the LFW for learning and evaluate it in the FERET experiment. Both datasets are very different from each other since the FERET set is captured in controlled conditions while the LFW is unconstrained. Table 8 shows the results of the cross-dataset experiment. It can be seen that although the performance is lessened as expected, it is still comparable to other state-of-the-art

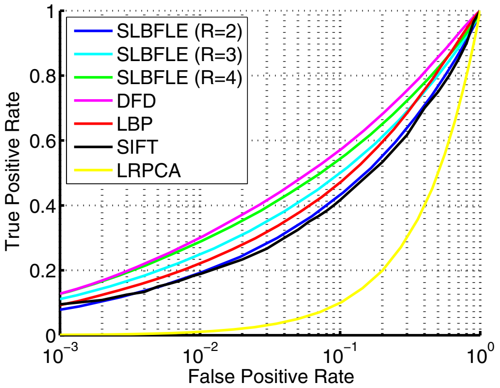


Figure 9. ROC curves of different feature descriptors on the PaSC dataset in the unsupervised setting.

Table 8. Cross Dataset experiment results showing rank-one recognition rates (%) for the FERET dataset, AUC and mean verification rate for the unsupervised and restricted setting of the LFW dataset, respectively.

Learn	Test	fb	fc	dup1	dup2
FERET	FERET	99.9	100.0	94.5	90.9
LFW	FERET	99.9	100.0	93.8	89.3
		AUC (Unsupervised)		Acc (Restricted)	
LFW	LFW	90.7		86.7 ± 1.30	
FERET	LFW	90.2		86.5 ± 1.85	

Table 9. Comparisons of the recognition performance of different variations of our methods, where the mean verification rate is used for LFW (unsupervised) and YTF,verification rate at false acceptance rate of 0.1% for PaSC, and the mean rank-1 identification of all the four subjects for FERET, respectively.

Dataset	SLBFLE1	SLBFLE2	SLBFLE3	SLBFLE
LFW	82.4	82.6	83.2	**84.2**
YTF	81.1	81.2	81.4	**82.4**
PaSC	27.7	27.5	28.3	**29.2**
FERET	97.3	97.4	97.6	**97.7**

methods.

Performance Analysis of Different Components in SLBFLE: We also investigated three other baselines (SLBFLE1, SLBFLE2 and SLBFLE3) of our SLBFLE to show the contribution of each term of the objective function. SLBFLE1 is a variation of our SLBFLE method without J_2, SLBFLE2 is another variation of our SLBFLE method without J_3, while SLBFLE3 is a variation without J_4. In this experiment, we use the SLBFLE features extracted at R=4. Table 9 shows the recognition performance of the different variations of our method on different datasets. We see that minimizing the quantization loss contributes more in the performance of our SLBFLE method. Nevertheless, J_3 also contributes to the performance as shown in the overall performance.

Table 10. Comparisons of local and global learning of the SLBFLE method in the LFW dataset with the unsupervised setting.

Method	Global SLBFLE	Local SLBFLE
Accuracy	84.18	**84.75**

Global SLBFLE vs. Local SLBFLE: Finally, we implemented a local learning method to further improve the SLBFLE method. In our current implementation, we learned binary codes and dictionary using random PDVs across the whole face image, so that the method is a global feature learning method because the position information of different face regions was ignored. In local SLBFLE, we learned individual binary codes and dictionaries for each face regions, which can exploit more local facial structure information for feature learning. In this experiment, we used an 8×8 block size so that 64 projections and dictionary matrices were learned. Similarly, we use the SLBFLE features extracted at R=4. Table 10 shows the recognition rate on LFW. We see that local SLBFLE can further improve the verification rate by 0.62%.

4. Conclusion

In this paper, we have proposed a new simultaneous local binary feature learning and encoding (SLBFLE) method for face recognition. Experiments on four benchmark face databases clearly demonstrate that our method achieved better or very competitive recognition performance with the state-of-the-art face feature descriptors. How to apply our proposed method to other computer vision applications such as object recognition and visual tracking seems to be an interesting future work.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grants 61225008, 61572271, 61527808, 61373074 and 61373090, 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. *PAMI*, 28(12):2037–2041, 2006. 1, 2, 6

[2] S. R. Arashloo and J. Kittler. Efficient processing of mrfs for unconstrained-pose face recognition. In *BTAS*, pages 1–8, 2013. 5

[3] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz. Fast high dimensional vector multiplication face recognition. In *ICCV*, pages 1960–1967, 2013. 6

[4] P. N. Belhumeur, J. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *PAMI*, 19(7):711–720, 1997. 1

[5] J. Beverdige, H. Zhang, B. Draper, P. Flynn, Z. Feng, P. Huber, J. Kittler, Z. Huang, S. Li, Y. Li, M. Kan, R. Wang, S. S, X. Chen, H. Li,

G. Hua, V. Struc, J. Krizaj, C. Ding, D. Tao, and P. Philips. Report on the fg 2015 video person recognition evaluation. In *FG*, pages 1–8, 2015. 7

[6] J. R. Beveridge, P. J. Phillips, D. S. Bolme, B. A. Draper, G. H. Given, Y. M. Lui, M. N. Teli, H. Zhang, W. T. Scruggs, K. W. Bowyer, et al. The challenge of face recognition from digital point-and-shoot cameras. In *BTAS*, pages 1–8, 2013. 7

[7] X. Cao, D. Wipf, F. Wen, G. Duan, and J. Sun. A practical transfer learning algorithm for face verification. In *ICCV*, pages 3208–3215, 2013. 6

[8] Z. Cao, Q. Yin, X. Tang, and J. Sun. Face recognition with learning-based descriptor. In *CVPR*, pages 2707–2714, 2010. 1, 2, 6

[9] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: high-dimensional feature and its efficient compression for face verification. In *CVPR*, pages 3025–3032, 2013. 5

[10] D. Cox and N. Pinto. Beyond simple features: A large-scale feature search approach to unconstrained face recognition. In *FG*, pages 8–15, 2011. 6

[11] 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 *CVPR*, pages 3554–3561, 2013. 6

[12] Y. Gong and S. Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In *CVPR*, pages 817–824, 2011. 4

[13] T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective face frontalization in unconstrained images. *arXiv preprint arXiv:1411.7964*, 2014. 6

[14] M.-V. Heydi, M.-D. Yoanna, and Z. Chai. Volume structured ordinal features with background similarity measure for video face recognition. In *ICB*, pages 1–6, 2013. 6

[15] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, pages 1875–1882, 2014. 5, 6

[16] J. Hu, J. Lu, J. Yuan, and Y.-P. Tan. Large margin multi-metric learning for face and kinship verification in the wild. In *ACCV*, pages 252–267. 2015. 6

[17] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, 07-49, UMass Amherst, 2007. 4

[18] G. B. Huang, H. Lee, and E. G. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*, pages 2518–2525, 2012. 6

[19] S. U. Hussain, T. Napoléon, F. Jurie, et al. Face recognition using local quantized patterns. In *BMVC*, pages 1–12, 2012. 1, 2, 7

[20] J. Kittler, A. Hilton, M. Hamouz, and J. Illingworth. 3d assisted face recognition: a survey of 3d imaging, modelling and recognition approachest. In *ECCV*, pages 469–481, 2004. 2, 7

[21] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng. ICA with reconstruction cost for efficient overcomplete feature learning. In *NIPS*, pages 1017–1025, 2011. 3

[22] H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. In *NIPS*, pages 801–808, 2006. 3

[23] Z. Lei, S. Z. Li, R. Chu, and X. Zhu. Face recognition with local gabor textons. In *ICB*, pages 49–57, 2007. 1

[24] Z. Lei, S. Liao, M. Pietikainen, and S. Z. Li. Face recognition by exploring information jointly in space, scale and orientation. *TIP*, 20(1):247–256, 2011. 1, 7

[25] Z. Lei, M. Pietikainen, and S. Z. Li. Learning discriminant face descriptor. *PAMI*, 6(4):1275–1286, 2013. 1, 2, 5, 6, 7

[26] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang. Probabilistic elastic matching for pose variant face verification. In *CVPR*, pages 3499–3506, 2013. 5, 6

[27] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt. Eigen-PEP for video face recognition. In *ACCV*, pages 1–16, 2010. 6

[28] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *TIP*, 11(4):467–476, 2002. 1

[29] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 7

[30] H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *ACCV*, pages 709–720, 2010. 6

[31] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *PAMI*, 22(10):1090–1104, 2000. 6

[32] H. J. Seo and P. Milanfar. Face verification using the lark representation. *TIFS*, 6(4):1275–1286, 2011. 5, 6

[33] C. Shan, S. Gong, and P. W. McOwan. Robust facial expression recognition using local binary patterns. In *ICIP*, volume 2, pages 914–917, 2005. 2

[34] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009. 2

[35] G. Sharma, S. ul Hussain, and F. Jurie. Local higher-order statistics (LHS) for texture categorization and facial analysis. In *ECCV*, pages 1–12, 2012. 5

[36] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, pages 1891–1898, 2014. 1

[37] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1–8, 2014. 6

[38] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991. 1

[39] R. Verschae, J. Ruiz-del Solar, M. Correa, et al. Face recognition in unconstrained environments: a comparative study. In *ECCVW*, pages 1–12, 2008. 5

[40] N.-S. Vu. Exploring patterns of gradient orientations and magnitudes for face recognition. *TIFS*, 8(2):295–304, 2013. 7

[41] N.-S. Vu and A. Caplier. Face recognition with patterns of oriented edge magnitudes. In *ECCV*, pages 313–326, 2010. 5

[42] N.-S. Vu and A. Caplier. Enhanced patterns of oriented edge magnitudes for face recognition and image matching. *TIP*, 21(3):1352–1365, 2012. 1, 7

[43] J. Wang, S. Kumar, and S.-F. Chang. Semi-supervised hashing for scalable image retrieval. In *CVPR*, pages 3424–3431, 2010. 3, 4

[44] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, pages 1–38, 2013. 4

[45] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, pages 529–534, 2011. 5, 6

[46] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *ECCVW*, 2008. 6

[47] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *ACCV*, pages 88–97, 2010. 6

[48] L. Wolf and N. Levy. The svm-minus similarity score for video face recognition. In *CVPR*, pages 3523–3530, 2013. 6

[49] A. Y. Yang, S. S. Sastry, A. Ganesh, and Y. Ma. Fast l 1-minimization algorithms and an application in robust face recognition: A review. In *ICIP*, pages 1849–1852, 2010. 3

[50] D. Yi, Z. Lei, and S. Z. Li. Towards pose robust face recognition. In *CVPR*, pages 3539–3545, 2013. 5, 6

[51] Y. Ying and P. Li. Distance metric learning with eigenvalue optimization. *JMLR*, 13(1):1–26, 2012. 6

[52] B. Zhang, Y. Gao, S. Zhao, and J. Liu. Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. *TIP*, 19(2):533–544, 2010. 1, 7

[53] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *TIP*, 16(1):57–68, 2007. 1, 7

[54] 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 *ICCV*, pages 786–791, 2005. 1, 7

[55] W.-Y. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM CSUR*, 35(4):399–458, 2003. 1

[56] 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 *CVPR*, pages 787–796, 2015. 6