Cost-Sensitive Local Binary Feature Learning for Facial Age Estimation

Jiwen Lu, *Senior Member, IEEE*, Venice Erin Liong, and Jie Zhou, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a cost-sensitive local binary feature learning (CS-LBFL) method for facial age estimation. Unlike the conventional facial age estimation methods that employ hand-crafted descriptors or holistically learned descriptors for feature representation, our CS-LBFL method learns discriminative local features directly from raw pixels for face representation. Motivated by the fact that facial age estimation is a cost-sensitive computer vision problem and local binary features are more robust to illumination and expression variations than holistic features, we learn a series of hashing functions to project raw pixel values extracted from face patches into low-dimensional binary codes, where binary codes with similar chronological ages are projected as close as possible, and those with dissimilar chronological ages are projected as far as possible. Then, we pool and encode these local binary codes within each face image as a real-valued histogram feature for face representation. Moreover, we propose a cost-sensitive local binary multi-feature learning method to jointly learn multiple sets of hashing functions using face patches extracted from different scales to exploit complementary information. Our methods achieve competitive performance on four widely used face aging data sets.**

***Index Terms*—Facial age estimation, feature learning, cost-sensitive learning, multi-feature learning, biometrics.**

I. Introduction

In RECENT years, facial age estimation has attracted much attention in computer vision and numerous facial age estimation methods have been presented in the literature [3], [5]–[7], [11], [12], [15], [16], [18], [21], [23], [24], [27], [32], [33], [36], [63]. The objective of facial age estimation is to predict the age value/group of a person of interest from his/her face image, which has wide potential applications such as human-computer interaction, soft biometrics, and social media analysis.

There are two key modules in a practical facial age estimation system: feature representation and age prediction. Representative facial feature representation methods include active appearance model (AAM) [15], [16], [33], Gabor wavelets [38], holistic subspace features [12], [18], local binary patterns (LBP) [1], and bio-inspired features (BIF) [24]. Having obtained the feature description for each face image, age prediction can be formulated as a classification [16], [24], [32], regression [12], [18], or ranking [5], [36] problem, respectively.

Most existing facial age estimation methods usually employ hand-crafted feature descriptors such as LBP and AAM for face representation, which require strong prior knowledge to engineer them by hand. There have also been some attempts on learning-based feature representation in facial age estimation [11], [12], [18], [21], [23], [24], which learn discriminative features directly from raw pixels. However, features learned by these methods are holistical so that they are not robust enough to local variations.

In this paper, we propose a cost-sensitive local binary feature learning (CS-LBFL) method for facial age estimation. Fig. 1 shows the pipeline of our proposed approach. In contrast to existing facial age estimation methods, our approach learns discriminative local face descriptor directly from raw pixel values for face representation. Specifically, we learn a series of hashing functions to project raw pixel values into low-dimensional binary codes so that codes with similar chronological ages are projected as close as possible and those with dissimilar chronological ages are projected as far as possible. Then, we pool and encode these local binary codes within a face image into a real-valued histogram feature for face representation. We also propose a cost-sensitive local binary multi-feature learning (CS-LBMFL) to learn multiple sets of hashing functions for face patches extracted from multiple scales to exploit complementary information to improve the performance. Experimental results on four widely used face aging datasets show the effectiveness of the proposed methods.

II. Related Work

In this section, we briefly review three related topics: 1) facial age estimation, 2) cost-sensitive learning, and 3) feature learning.

A. Facial Age Estimation

Recent years have witnessed a considerable interest in facial age estimation [5]–[7], [11], [12], [15], [16], [18], [21], [24], [27], [32], [33], [36], [63]. For example, Lanitis *et al.* [33] proposed a quadratic function with AAM for age regression. Yan *et al.* [61] presented a semi-definite programming

Manuscript received March 2, 2015; revised August 4, 2015; accepted September 12, 2015. Date of publication September 23, 2015; date of current version October 6, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61225008, Grant 61572271, Grant 61527808, Grant 61373074, and Grant 61373090, in part by the National Basic Research Program of China under Grant 2014CB349304, in part by the Ministry of Education of China under Grant 20120002110033, and in part by the Tsinghua University Initiative Scientific Research Program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. A. N. Rajagopalan.

J. Lu and J. Zhou are with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: elujiwen@gmail.com; jzhou@tsinghua.edu.cn).

V. E. Liong is with the Rapid-Rich Object Search Laboratory, Interdisciplinary Graduate School, Nanyang Technological University, Singapore 639798 (e-mail: veniceer001@e.ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2015.2481327

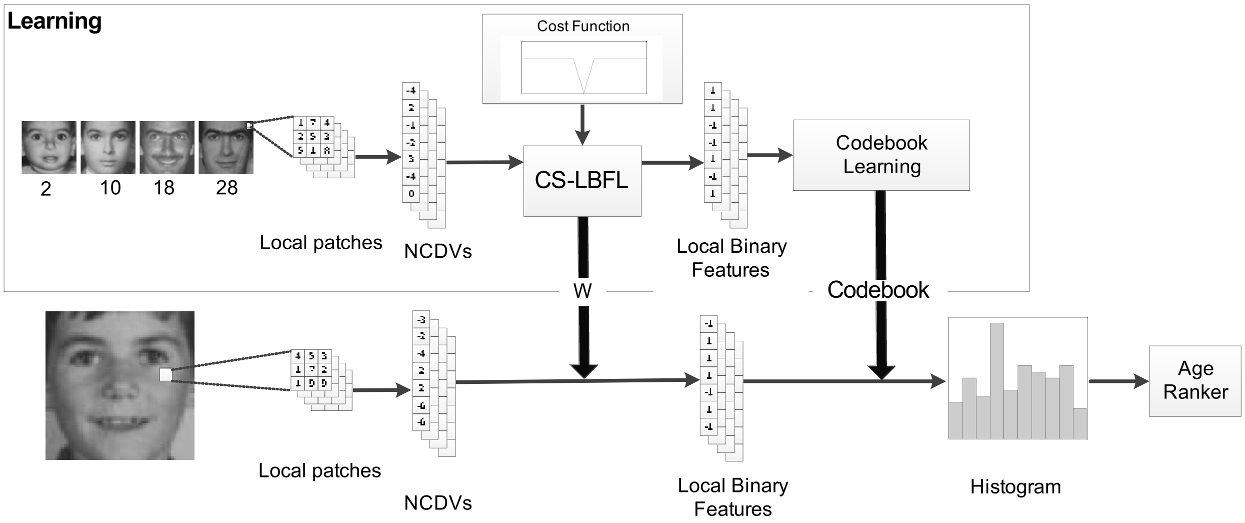


Fig. 1. Pipeline of the proposed facial age estimation approach. In the training phase, we extract a neighbor-center difference vector (NCDV) for each pixel in the original face image and learn a projection matrix W to map these NCDVs into low-dimensional binary codes, where a cost function is applied to reflect the age relationship of different age classes. Then, we cluster these binary codes into a codebook and encode them within a face image into a real-valued histogram feature for face representation. Lastly, we train an age ranker using these histogram features of the training samples. In the testing phase, for each given testing face image, we first extract all NCDVs within the image and project them into binary codes using the learned projection matrix W. Then, we encode these binary codes within the same face as a histogram feature vector. Finally, the age is predicted using the learned age ranker.

regression model with nonnegative label intervals. Montillo and Ling [45] performed age regression from features selected by random forests. Zhang and Yueng [63] presented a multi-task Gaussian process method to learn person specific age estimator. Chang *et al.* [4] proposed an ordinal ranking method by casting the age estimation problem as a series of binary classification subproblems. Geng *et al.* [15] presented a label distribution learning method to model the relationship of face samples and age labels. Guo and Mu *et al.* [20], [22] used biologically inspired features and exploited the information of human gender and race for facial age estimation. While these methods have achieved reasonably good performance, most of them use hand-crafted features for face representation, which require strong priors to engineer these descriptors by hand. More recently, Guo *et al.* [18], [23], [24], Guo and Mu [21], Fu *et al.* [11], and Fu and Huang [12] proposed several holistic feature learning methods using discriminative manifold learning techniques. However, these methods learn feature descriptors holistically so that they are not robust to local variations. In contrast to these previous works, we propose a feature learning approach to learn discriminative local binary face descriptor directly from raw face images.

B. Cost-Sensitive Learning

Cost-sensitive learning is an important topic in data mining and machine learning, and many cost-sensitive learning algorithms have been proposed in the literature [8], [10], [37], [39], [42], [47], [51], [53], [62], [64]. For cost-sensitive learning, cost information of different samples is utilized to characterize their importance to reflect different amounts of losses in a classification system. Representative cost-sensitive learning methods include cost-sensitive boosting [42], [51], cost-sensitive support vector machine [63], cost-sensitive neural networks [47], cost-sensitive subspace analysis [40], [41], cost-sensitive semi-supervised learning [37], cost-sensitive subspace learning [40], [41], and cost-sensitive feature selection [43]. Facial age estimation is a typical cost-sensitive computer vision problem because mis-estimating face samples with 20 years old as 30 years old incurs higher loss than that as 25 years. Motivated by this, Chang *et al.* [5] presented a cost-sensitive ordinal ranking approach for facial age estimation, where the cost-sensitive information is exploited in ranking model. Even if they have achieved encouraging performance, they only utilized the cost-sensitive information in the age prediction stage because they employed AAM and LBP features for facial feature representation. In this work, we propose a cost-sensitive local feature learning approach for age estimation, where the cost information is exploited in the feature extraction stage. Hence, our approach is complementary to the existing cost-sensitive learning methods.

C. Feature Learning

A number of feature learning methods have been proposed in recent years [2], [28], [29]. Representative feature learning methods include sparse auto-encoder [2], restricted Boltzmann machine [28], convolutional neural networks [29], denoising auto-encoders [49], and reconstruction independent component analysis [34]. Among these feature learning methods, convolutional neural networks [29], [52] has achieved the superb performance in various computer vision tasks. However, a large number of labeled samples are required for convolutional neural networks to train the model because there are usually extensive parameters to estimate. For facial age estimation, it is difficult to collect such large number of labeled training data. Hence, it is desirable to learn discriminative features with limited number of training samples. Since facial age estimation is a cost-sensitive computer vision problem, it is also

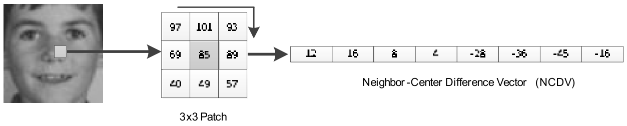


Fig. 2. Illustration of extracting a NCDV within a face patch. For a face patch of size $n \times n$, the neighbor to center difference vector (NCDV) is extracted by subtracting the neighboring pixels to the center pixel, which is a $(n^2 - 1)$-dimensional feature vector. In this figure, n is selected as 3 and hence the extracted NCDV is an 8D feature vector.

desirable to exploit such cost information in discriminative feature extraction. Moreover, most existing feature learning methods only learn real-valued features from raw data, which have been proven to be less robust to expression and illumination variations than local binary features because the quantized binary codes can alleviate these variations [1], [26], [46], [50]. In this work, we learn local binary discriminative features for facial age estimation in a cost-sensitive manner.

III. Proposed Approach

In this section, we first detail the proposed CS-LBFL method and then present the extended CS-LBMFL method.

A. Cost-Sensitive Local Binary Feature Learning

Unlike most existing feature learning methods [29], [34] which use the original raw pixel patch to learn feature representations, we learn discriminative features using the neighbor-centroid difference vectors (NCDV). NCDV computes the difference between the center point and neighboring pixels within a patch so that it better describes how pixel values change and implicitly encode important visual patterns such as edges and lines in face images than raw pixels. Moreover, NCDV has been widely used in many previous local face descriptors, such as hand-crafted LBP [31] and learning-based DFD [35]. Fig. 2 illustrates how to extract one NCDV from a face patch.

Let $X = [x_1, \ldots, x_n, \ldots x_N] \in R^{d \times N}$ be the training set, where x_n $(1 \leq n \leq N)$ is the nth NCDV and d is the feature dimension of each NCDV, respectively. We aim to seek a mapping to project each NCDV into p bits of binary codes, where the ith bit is computed as follows:

$$b_{ni} = \mathrm{sgn}(w_i^T x_n), \tag{1}$$

where w_i is the projection vector for the ith bit, $\mathrm{sgn}(v)$ equals to 1 if $v \geq 0$ and -1 otherwise.

By combining all w_i $(1 \leq i \leq p)$ into a projection matrix $W = [w_1, w_2, \cdots, w_p] \in R^{d \times p}$, we map each NCDV into a binary feature vector $b_n \in R^p$ as follows:

$$b_n = \mathrm{sgn}(W^T x_n). \tag{2}$$

For each NCDV x_n in the training set, we select two other NCDV vectors x_n^+ and x_n^- to construct one positive pair and one negative pair to learn the projection matrix W, where x_n, x_n^+ and x_n^- are NCDV feature vectors extracted at the same position from different face images, x_n and x_n^+ are from face images in the same age class, and x_n and x_n^- are from face images in different age classes. Assume there are M NCDV face pairs in the training set and the mth face pair is represented as $(x_{1m}, x_{2m}, y_{1m}, y_{2m}, \ell_m)$, where x_{1m} and x_{2m} are the NCDV vectors in this pair, y_{1m} and y_{2m} are the corresponding age labels, ℓ_m is a flag number which is set to 1 if this pair is positive and -1 otherwise.

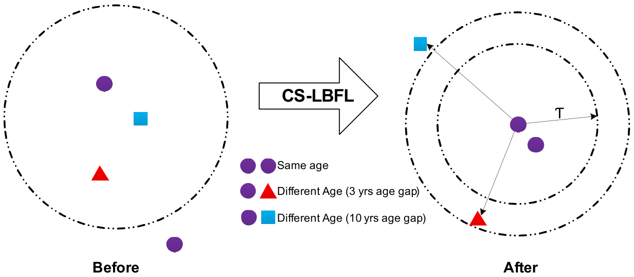


Fig. 3. The basic idea of our cost-sensitive feature learning approach. There are 4 NCDV samples in the training set, where two of them form a positive pair (two circles) and three of them form two negative pairs. The first negative pair have a large age gap (the circle and the square) and the other negative pair have a small age gap (the circle and the triangle). We learn the projection matrix W so that the distance of positive pairs is smaller than a threshold and those of negative pairs are larger than the threshold, and the negative pair with larger age gap has larger distance than those with smaller age gap in the learned feature space.

Our proposed CS-LBFL method aims to learn l discriminative hash functions to obtain a binary feature vector for each NCDV, which is formulated as the following optimization problem:

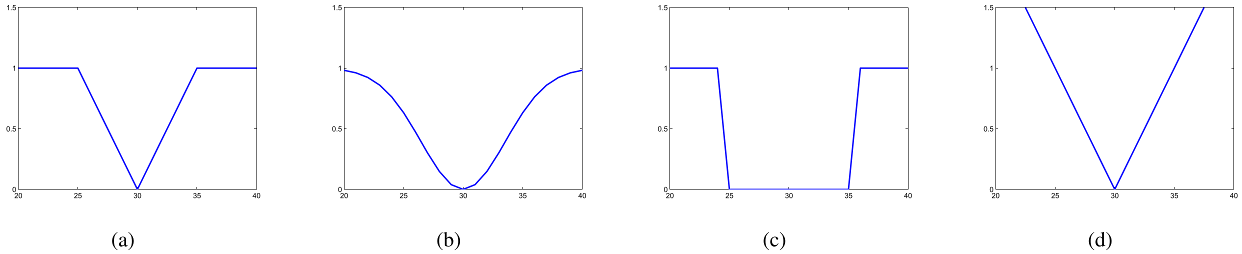
$$\min_W J = \sum_{m=1}^{M} (1 - \ell_m(\tau - d(b(x_{1m}), b(x_{2m}))Q(y_{1m}, y_{2m})), \tag{3}$$

where $d(b(x_{1m}), b(x_{2m}))$ is the hamming distance between the binary code of x_{1m} and x_{2m}, $b(x_{1m})$ and $b(x_{2m})$ are computed according to (2), τ is a pre-specified parameter, $Q(y_{1m}, y_{2m})$ is the cost information to reflect the relative relationship of binary codes from different classes. Fig. 3 shows the basic idea of our cost-sensitive feature learning approach.

There are two key objectives for (3):

1) The distance between the learned binary codes of two NCDV vectors from the same age class is expected to be less than a threshold and that from different classes is higher than the threshold, so that the margin between positive pairs and negative pairs is maximized and discriminative information is exploited in the learned binary codes [55].
2) The difference between binary codes from a negative pair with a small age gap is smaller than that from a negative pair with a large age gap because the estimation error of a negative pair with smaller gap is less than that with a larger gap. Hence, different weights should be assigned to different negative pairs with different age gaps.

Generally, there are a number of possible functions which can be used to exploit the cost information in different age classes. In our work, we exploit the following



(a) (b) (c) (d)

Fig. 4. Illustration of different cost functions to define the matrix $Q(c_1, c_2)$, where (a) linear-truncated, (b) Gaussian-truncated, (c) truncated and (d) linear functions are used, respectively. In these examples, c_1 is set as 20 and L is set as 5. When c_2 is near to c_1, the cost value is small because mis-estimating samples in the c_1th class as the c_2th class occur a small error.

four cost functions:

$$Q(c_1, c_2) = \begin{cases} \frac{|c_2 - c_1|}{L}, & |c_1 - c_2| \leq L \\ 1, & \text{otherwise} \end{cases}$$

$$Q(c_1, c_2) = \begin{cases} 1 - \exp^{\frac{-(c_2 - c_1)^2}{L^2}}, & |c_1 - c_2| \leq L \\ 1, & \text{otherwise} \end{cases}$$

$$Q(c_1, c_2) = \frac{|c_2 - c_1|}{L}$$

$$Q(c_1, c_2) = \begin{cases} 0, & |c_1 - c_2| \leq L \\ 1, & \text{otherwise} \end{cases}$$

where L is a constant parameter that describes the tolerance level of varying age relationships. Fig. 4 illustrates these four cost functions to show how they exploit different relations for neighboring age classes.

While the cost functions in our method are similar to those used in robust statistics, there are two key difference between them:

1) Their physical meanings are intrinsically different because the value $Q(c_1, c_2)$ of the cost functions in robust statistics is to measure the probability of the sample from the c_1th class to the c_2th class while that in our method is to measure the loss of mis-estimating samples from the c_1th class as the c_2th class.
2) Their objectives are different because the cost functions in robust statistics are used for parameter estimation while those in our method are employed for feature learning.

Since (3) is an NP-hard problem because of the non-linear sgn(·) function, we relax the binary term by replacing the sign of projection with its signed magnitude [17], [54]. Then, the objective function of CS-LBFL can be rewritten as follows:

$$\min_{W} J = \sum_{m}^{M} (1 - \ell_m(\tau - d^2(x_{1m}, x_{2m})) \times Q(y_{1m}, y_{2m}) \quad (4)$$

where

$$\begin{aligned} d^2(x_{1m}, x_{2m}) &= \|W^T x_{1m} - W^T x_{2m}\|_F^2 \\ &= W^T (x_{1m} - x_{2m})(x_{1m} - x_{2m})^T W \quad (5) \end{aligned}$$

```
Algorithm 1 CS-LBFL
    Input: Training set: $M$ NCDV face patch pairs,
           iteration number: $T$, learning rate: $\eta$.
    Output: Feature projection matrix: $W$.
    Step 1: Initialize $W = W^0$.
    Step 2: Update $W$ using (6).
    Step 3: Output feature projection matrix: $W = W^t$.
```

To solve the optimization problem in (4), we use the stochastic sub-gradient descent scheme to obtain the parameter W in an iterative manner. At each iteration, we sample a pair of feature patches and update W as follows:

$$W^{t+1} = \begin{cases} W^t - \eta \ell_m W^t \delta_m q_m, & \text{if } \ell_m = -1, \\ W^t - \eta \ell_m W^t \delta_m, & \text{if } \ell_m = 1, \\ W^t, & \text{otherwise} \end{cases} \quad (6)$$

where $q_m = Q(y_{1m}, y_{2m})$, which is determined from the cost matrix, $\delta_m = (x_{1m} - x_{2m})(x_{1m} - x_{2m})^T$ is the outer product of x_{1m} and x_{2m}, $\eta > 0$ is the learning rate. In our implementations, W_0 is initialized by selecting the p largest eigenvectors of the PCA subspace which is learned from these NCDV vectors. **Algorithm 1** summarizes the proposed CS-LBFL method.

B. Cost-Sensitive Local Binary Multi-Feature Learning

While CS-LBFL learns discriminative features from raw face patches, only a single scale face patch is used for feature learning. Previous studies have shown that face patches extracted from multiple scales provide complementary information for discriminative feature extraction [41]. Hence, it is desirable to extract multiple NCDVs to learn discriminative features from different scales to extract complementary information to improve the performance. A naive solution is to combine multiple NCDVs into a longer NCDV feature vector and then apply CS-LBFL for feature learning with these concatenated NCDVs. However, this operation is suboptimal because each NCDV has a specific statistical characteristic and such a concatenation sacrifices the diversity of different descriptors. To this end, we also propose a cost-sensitive local binary multi-feature learning (CS-LBMFL) method to jointly learn multiple feature projection matrices for feature learning. Specifically, one feature projection matrix is learned for each size of NCDV under which the characteristic of CS-LBFL is preserved in each NCDV feature space and the interaction of different feature projection matrices is also exploited, simultaneously.

Assume there are K NCDV feature vectors extracted at each position from different scales in each face image.

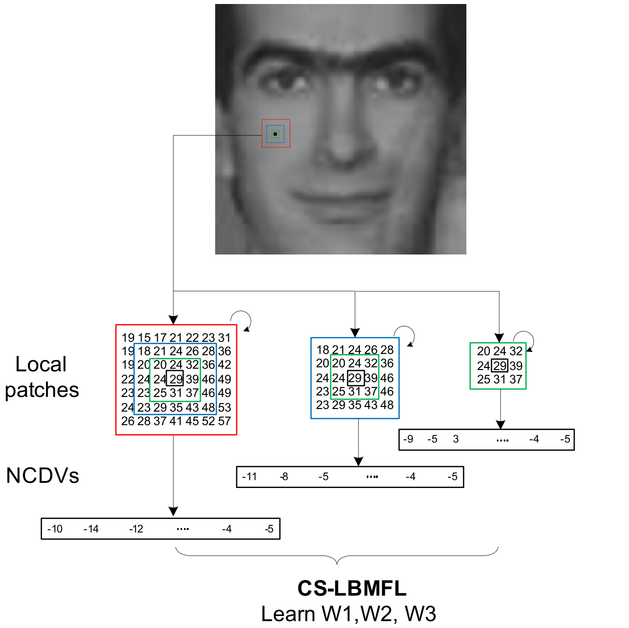


Fig. 5. To perform multi-feature learning, we extract multiple NCDVs of different sizes at the same position. In this figure, we extract three different sizes of NCDV feature vectors at the same position, whose sizes are 5 × 5, 7 × 7, and 9 × 9, respectively. Hence, three corresponding NCDV feature vectors are obtained, and their dimensions are 24, 48, and 80, respectively. For each size of NCDV feature vector, we learn one projection to map it into a binary feature vector. In our CS-LBMFL method, these multiple projections are jointly learned so that both discriminative and complementary information is effectively exploited.

For the kth NCDV feature space, there are M face pairs, $1 \leq k \leq K$, and the mth pair of face patches for the kth feature is represented as $(x^k_{1m}, x^k_{2m}, y_{1m}, y_{2m}, \ell_m)$, where x^k_{1m} and x^k_{2m} are the face pair of NCDV feature vectors in the kth NCDV feature space, y_{1m} and y_{2m} are the corresponding age labels, ℓ_m is a flag number which is set to 1 if the face pair is positive and -1 otherwise. Fig. 5 shows how to extract multiple NCDV feature vectors from different scales at the same position for feature learning.

Similar to CS-LBFL, CS-LBMFL aims to jointly learn K feature projection matrices under which the distance of each positive pair in the corresponding feature space is less than a threshold and that of each negative pair is higher than a threshold, and the difference of feature representations of each position across NCDV feature spaces is minimized because the NCDV feature vectors share the same semantic label, so that both the discriminative information and complementary information can be simultaneously exploited. To achieve this, we formulate our CS-LBMFL method as the following optimization problem:

$$\min_{W_1,\cdots,W_K,\alpha} \quad H = \sum_{k=1}^{K} \alpha_k J(W_k) + \lambda G(W_1, \cdots, W_k)$$
$$\text{subject to } \sum_{k=1}^{K} \alpha_k = 1, \alpha_k \geq 0. \tag{7}$$

where

$$f_k(W_k) = \sum_{m}^{M} (1 - \ell_i(\tau - d^2(b(x^k_{1m}), b(x^k_{2m}))) \times Q(y_{1m}, y_{2m}) \tag{8}$$

$$G(W_1, \cdots, W_K) = \sum_{\substack{k_1,k_2=1\\ k_1 \neq k_2}}^{K} \sum_{m}^{M} (d(b(x^{k_1}_{1m}), b(x^{k_2}_{1m})) + d(b(x^{k_1}_{2m}), b(x^{k_2}_{2m}))) \tag{9}$$

W_k is the feature projection matrix for the kth NCDV feature, λ is a parameter to balance these two terms, $\alpha = [\alpha_1, \cdots, \alpha_K]$ is the weighting vector and α_k is the weight of the kth NCDV feature, $\alpha_k \geq 0$.

There are two objectives in (7):

1) The distance between the learned binary codes of each pair of NCDV vectors is optimized with the CS-LBFL criterion in each single NCDV feature space.
2) The difference between the learned binary codes of different NCDVs extracted at the same position is minimized in the jointly learned feature spaces.

Similar to CS-LBFL, we relax the binary term to a signed-magnitude vector. To our best knowledge, there is no closed-form solution to (7) because we need to solve the weighting vector α and K feature projection matrices $W_1, \cdots, W_K$ simultaneously. To address this, we propose an alternating optimization algorithm to obtain a local optimal solution. Specifically, we first initialize $W_1, \cdots, W_{k-1}, W_{k+1}, \cdots, W_K$ and α and solve W_k sequentially, and then update α with the learned $W_1, \cdots, W_K$ accordingly.

When $W_1, \cdots, W_{k-1}, W_{k+1}, \cdots, W_K$ and α are fixed, (7) can be rewritten as follows:

$$\min_{W_k} H(W_k) = \alpha_k J_k(W_k) + \lambda G_k(W_k) \tag{10}$$

where

$$G(W_k) = \sum_{l=1, l \neq k}^{K} \sum_{m=1}^{M} (\| W_k^T x^k_{1m} - W_l^T x^l_{1m} \|_F^2 + \| W_k^T x^k_{2m} - W_l^T x^l_{2m} \|_F^2) \tag{11}$$

The stochastic sub-gradient descent scheme is used to obtain the parameter W_k^t iteratively as follows:

$$W_k^{t+1} = W_k^t - \eta \left(\alpha_k \frac{\partial f_k(\partial W_k^t)}{W_k^t} + \lambda \frac{\partial G(W_k^t)}{\partial W_k^t} \right) \tag{12}$$

where $\eta > 0$ is the learning rate, and

$$\frac{\partial J_k(W_k^t)}{\partial W_k^t} = \begin{cases} \ell_m W_k^t \delta_m q_m, & \text{if } \ell_m = -1 \\ \ell_m W_k^t \delta_m, & \text{if } \ell_m = 1 \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

$$\frac{\partial G_k(W_k^t)}{\partial W_k^t} = 2\lambda(K-1) W_k^t \sum_{l=1, l \neq k}^{K} \sum_{m=1}^{M} (x^l_{1m})^T x^l_{1m} + 2\lambda(K-1) W_k^t \sum_{l=1, l \neq k}^{K} \sum_{m=1}^{M} (x^l_{2m})^T x^l_{2m} - 2\lambda W_k^t \sum_{l=1, l \neq k}^{K} \sum_{m=1}^{M} (x^l_{1m})^T x^l_{1m} - 2\lambda W_k^t \sum_{l=1, l \neq k}^{K} \sum_{m=1}^{M} (x^l_{2m})^T x^l_{2m} \tag{14}$$

Algorithm 2 CS-LBMFL

```
Input: Training set $S = [S^k]$ and the iteration number $T$.
Output: Projection matrices $W_1, \ldots, W_K$ and the weighting vector $\alpha$.
Step 1 (Initialization):
    1.1. Initialize $W_k$ by selecting the $D$ largest eigenvectors of the PCA subspace which is learned from the NCDV vectors in the $k$th feature, and $\alpha = [1/K, \ldots, 1/K]$.
Step 2 (Local Optimization):
    For $t = 1, 2, \cdots, T$, repeat
        2.1. For $k = 1, \cdots, K$, Update $W_k^t$ using (12).
        2.2. Update $\alpha$ using (20).
Step 3 (Output projection matrices):
    3.1. Output projection matrices $W_k = W_k^t$.
```

Having obtained $W_1, \cdots, W_K$, we can obtain α by solving the following objective function:

$$\min_{\alpha} \; H = \sum_{k=1}^{K} \alpha_k J_k(W_k)$$
$$\text{subject to} \sum_{k=1}^{K} \alpha_k = 1, \quad \alpha_k > 0 \tag{15}$$

The trivial solution to (15) is $\alpha_k = 1$, which corresponds to the minimum $J_k(W_k)$ over different NCDV features, and $\alpha_k = 0$ otherwise. This means that only the best NCDV is selected and the complementary property of multiple NCDVs cannot exploited. To address this, we modify α_k to be α_k^r $(r > 1)$, and rewrite the following objective function:

$$\min_{\alpha} \; H = \sum_{k=1}^{K} \alpha_k^r J_k(W_k)$$
$$\text{subject to} \sum_{k=1}^{K} \alpha_k = 1, \alpha_k > 0 \tag{16}$$

We construct the following Lagrange function:

$$H(\alpha, \beta) = \sum_{k=1}^{K} \alpha_k^r J_k(W_k) - \beta(\sum_{k=1}^{K} \alpha_k - 1) \tag{17}$$

Let $\frac{\partial H(\alpha,\beta)}{\partial \alpha_k} = 0$ and $\frac{\partial H(\alpha,\beta)}{\partial \beta} = 0$, we have

$$r\alpha_k^{r-1} J_k(W_k) - \zeta = 0 \tag{18}$$
$$\sum_{k=1}^{K} \alpha_k - 1 = 0 \tag{19}$$

Combining (18) and (19), we update α_k as follows:

$$\alpha_k = \frac{(1/J_k(W_k))^{1/(r-1)}}{\sum_{k=1}^{K} (1/J_k(W_k))^{1/(r-1)}} \tag{20}$$

Algorithm 2 summarizes the proposed CS-LBMFL method.

C. Discussion

In this subsection, we highlight the difference between our cost-sensitive local binary feature learning model and several recently proposed methods.

1) Cost-Sensitive Subspace Learning and Cost-Sensitive Feature Selection [40], [41], [43]: In our recent work, we introduced cost-sensitive subspace learning for face recognition. The basic idea of cost-sensitive subspace learning is to learn a feature projection matrix to map each face sample from the original space into the feature space so that the cost-sensitive information can be preserved. More recently, Miao *et al.* [43] proposed a cost-sensitive feature selection method to select the most important features which yield the minimal loss for pattern classification. However, both of them are holistic feature learning approach because each sample is considered as a whole feature vector and the most informative features or subspaces are learned globally. In our CS-LBFL and CS-LBMFL methods, we exploit cost-sensitive information in raw face patches and learn local binary feature descriptor locally so that it is more robust to local variations in face images because our model inherits the advantage of LBP.

2) Discriminative Manifold Feature Learning [24]: Recently, Guo *et al.* [18], [23], [24], Guo and Mu [21], Fu *et al.* [11], and Fu and Huang [12] proposed several discriminative manifold feature learning methods for facial age estimation, where discriminative manifold features and the age information are modeled by a regressor. However, these methods learn feature descriptors holistically so that they are not robust to local variations. In contrast to these previous works, our feature learning approach learns local binary feature representation directly from raw face images, so that it is more robust to variations of illumination and expression.

IV. Experiments

We evaluate our CS-LBFL and CS-LBMFL on the widely used FG-NET [33], MORPH (Album 2) [48], LifeSpan [44], and FACES [9] datasets. The following describes the details of the experiments and results.

A. Experimental Settings

For CS-LBFL, we extracted each NCDV feature vector from a 7 × 7 local patch. Hence, each extracted NCDV is a 48D feature vector. We learned the project matrix W to map each NCDV into a p-dimensional binary feature vector. In our experiments, p was set as 15. Having obtained the binary codes for each NCDV, we clustered them to learn a codebook using K-means and pooled them to represent each image as a histogram feature. Previous studies have shown different face regions have different structural information [58] and it is desirable to learn position-specific features for face representation. Motivated by this, we divided each face image into 8 × 8 non-overlapped local regions and learn a CS-LBFL feature descriptor for each local region. Lastly, histogram features extracted from different regions are concatenated as the final representation for the whole face image. In our experiments, the codebook size, learning rate η, and threshold τ were empirically set as 500, 0.000001 and 2, respectively. Fig. 6 illustrates how to use CS-LBFL for face representation.

For CS-LBMFL, we extracted three sets of patches for each pixel at the same position. The patch sizes were set as 9 × 9, 7×7, and 5×5, which yield three NCDV feature vectors which

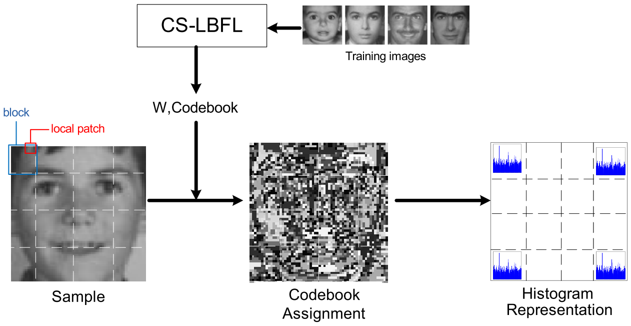


Fig. 6. Illustration of the face feature representation procedure in CS-LBFL. Having learned the codebook and parameter W in the training stage, we perform histogram representation for each face region. Specifically, each NCDV is then converted to local binary features using our method and is then converted as a real-valued histogram feature for face representation. In our method, the whole face is divided into several local regions and each patch is extracted from each local region to obtain the NCDV. In this figure, we use 4 × 4 blocks for easy of presentation. In our experiments, each face image was divided into 8 × 8 non-overlapped blocks.

are 80-, 48-, and 24D, respectively. Then, each NCDV feature set was clustered to form a codebook, which follows the same procedure in CS-LBFL. The binary codes length p, codebook size and parameter λ was empirically set as 15, 500, and 0.1, respectively,

Having obtained the feature representation of each face image, we used the ordinal hyperplanes ranker (OHRank) [5] as the age estimator because it has shown excellent performance in previous facial age estimation studies [5].

We employed two widely used measures for performance evaluation: 1) mean absolute error (MAE) [7], [12], [16], [18], [21], [63], and 2) the cumulative score (CS) [7], [12], [16], [18], [21], [63], which are defined as follows:

$$MAE = \sum_{i=1}^{N_{ts}} |\hat{l}_i - l_i| / N_{ts}$$

$$CS(\theta) = N_{e\leq\theta} / N_{ts} \times 100\%$$

where $\hat{l}_i$ and l_i are the predicted and original age labels of the ith testing sample, N_{ts} is the testing sample number, $N_{e\leq\theta}$ is the number of testing samples whose absolute errors are less than θ years old.

B. Experiments on the FG-NET Dataset

There are 1002 face images from 82 persons in the FG-NET face dataset [33]. Each person has 12 face images on average, and the age range is from 0 to 69 years old. There are large variations in pose, illumination, and expression. Fig. 7 shows some face examples from the FG-NET dataset. For each face image, we manually cropped and aligned it into 64×64 according to the eye positions. For color images in this dataset, we first converted them into gray-scale ones and then applied our feature learning methods for feature extraction.

We adopted the leave-one-person-out (LOPO) strategy to conduct age estimation experiments. Specifically, face images of one person were used as the test set and those of other persons were used for training. Finally, the average result over all 82 folds was used as the final age estimation performance.

Fig. 7. Several example facial images of one person in the FG-NET dataset, where the number below each image is the age value.

TABLE I

MAEs (Years Old) and CS (%) of CS-LBFL When Different Cost Functions Are Employed on the FG-NET Dataset

Cost function	MAE	CS ($\theta = 5$)
Linear-Truncate	**4.43**	**74.6**
Gaussian-Truncate	4.51	73.8
Truncate	4.54	73.4
Linear	4.47	74.3

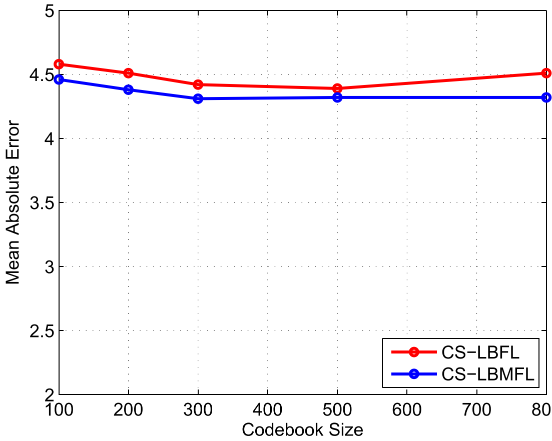


Fig. 8. MAE (years old) of our proposed methods versus different codebook sizes on the FG-NET dataset.

1) Parameter Determination: We first investigated different cost functions of our CS-LBFL. Table I shows the MAE and CS of our CS-LBFL with different cost functions. We see that the linear-truncate cost function achieves the best performance than other functions. Compared with other cost functions, the linear-truncate function exploits the cost-sensitive information on neighboring age labels, which is reasonable because when the label difference is large, the influence of such age difference is limited because there is no much difference between the age gap of 25 years old and 26 years old. Hence, we used the linear-truncate cost function for performance evaluation in the following experiments.

We also investigated the performance of our methods versus different codebook sizes. We varied the codebook size from 100 to 800. Fig. 8 shows the MAEs of our CS-LBFL and CS-LBMFL versus different codebook sizes. We see that our methods are not sensitive to the codebook size and the best estimation performance is obtained when the codebook size was set in the range of [300, 500]. In the following experiments, we set the codebook size as 500 for performance evaluation.

TABLE II

MAEs (Years Old) Comparison of Different Age Estimation Methods on the FG-NET Dataset

Method	MAE	Method description	year
KNN	8.24		
SVM	7.25		
MLP	6.95		
RUN [60]	5.78	AAM + RUN	2007
AGES [16]	6.77	AAM + Aging pattern subspace	2007
LARR [18]	5.07	AAM + Locally adjusted robust regression	2008
PFA [19]	4.97	AAM + Probabilistic fusion approach	2008
KAGES [14]	6.18	AAM + Kernel AGES	2008
MSA [13]	5.36	AAM + Multilinear subspace analysis	2009
SSE [59]	5.21	AAM + submanifold embedding	2009
mKNN [57]	5.21	AAM + metric learning	2009
MTWGP [63]	4.83	AAM + Multi-task warped GPR	2010
RED-SVM [4]	5.21	AAM + Red SVM	2010
OHRank [5]	4.48	AAM + Ordinal hyperplanes ranker	2011
PLO [36]	4.82	Feature selection + OHRank	2012
IIS-LLD [15]	5.77	AAM/BIF+learning from label distribution	2013
CPNN [15]	4.76	AAM/BIF+learning from label distribution	2013
CA-SVR [6]	4.67	AAM+cumulative/joint attribute learning	2013
CS-LBFL	**4.43**	Feature learning + OHRank	
CS-LBMFL	**4.36**	Multiple feature learning + OHRank	

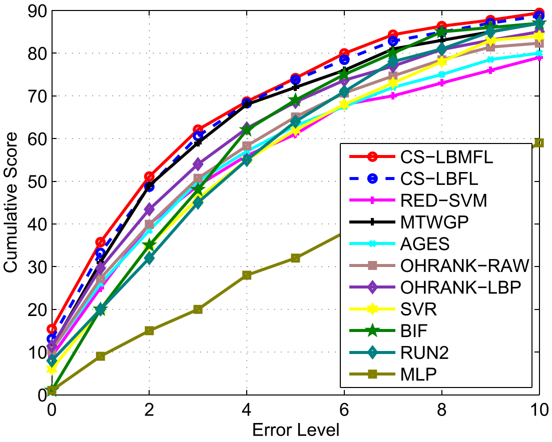


Fig. 9. CS curve comparison of different age estimation methods on the FG-NET dataset.

TABLE III

MAEs (Years Old) Comparison With Existing Feature Learning Methods on the FG-NET Dataset

Method	MAE
LQP [30]	4.70
DFD [35]	4.57
RICA [34]	6.09
CS-LBFL	**4.43**
MDFD	5.35
MLQP	4.61
MRICA	5.65
CS-LBMFL	**4.36**

2) Comparisons With State-of-the-Art Age Estimation Methods: Table II tabulates the MAEs of our CS-LBFL and CS-LBMFL on the FG-NET dataset, compared with state-of-the-art facial age estimation methods. Fig. 9 shows the CS curves of different facial age estimation methods. Results of the existing state-of-the-art methods are obtained from the original papers. As can be seen, our proposed CS-LBFL achieves competitive performance with the existing state-of-the-art facial age estimation methods. Moreover, the performance of our CS-LBFL can be further improved when it is extended to CS-LBMFL. This is because multiple feature information is exploited in CS-LBMFL, which can extract more complementary information to improve the estimation performance.

3) Comparisons With Existing Feature Learning Methods: We compared our CS-LBFL and CS-LBMFL with three existing feature learning methods including LQP [30], DFD [35] and RICA [34]. These methods have been successfully applied in face and object recognition in recent years [30], [34], [35]. In this work, we applied them for face feature learning in our age estimation task for comparison. The source codes of DFD and RICA are publicly available. We implemented LQP by carefully following the details of the paper. For fair comparison, we carefully tuned the parameters of the compared methods to achieve the best result. Specifically, for DFD and RICA, the patch size was set as 7 × 7 to learn the features and the codebook size was set as 500 to learn the dictionary. For LQP, the patch size was set as 7×7 and the threshold was set as 6 to learn two binary codes sets. Then, we learned two codebooks using K-means, where each codebook size was selected as 250. For all these three compared methods, the OHRank [5] age estimator was used for age prediction. Table III shows the MAEs of different feature learning methods. As can be seen, our proposed CS-LBFL outperforms LQP and RICA and achieves comparable performance with DFD.

To fairly compare our CS-LBMFL with these feature learning methods, we also performed feature learning using multiple scales for LQP, RICA and DFD, where multiple scales of local patches are used for feature learning. Specifically, we extended them into multi-scale LQP (MLQP), multi-scale (MDFD) and multi-scale (MRICA) by following

TABLE IV

MAEs (Years Old) Comparison With Existing Cost-Sensitive Learning Methods on the FG-NET Dataset

Method	MAE
CS-LDA [40]	7.36
CS-FS [43]	7.41
CS-LBFL	**4.43**
CS-LBMFL	**4.36**

TABLE V

MAEs (Years Old) Comparison With Different Age Estimators on the FG-NET Dataset

Method	MAE
CS-LBFL + SVR	4.98
CS-LBMFL + SVR	4.73
CS-LBFL + OHRank	**4.43**
CS-LBMFL + OHRank	**4.36**

the same extension procedure from CS-LBFL to CS-LBMFL. The age estimation performance of different feature learning methods are also shown in Table III. We see that our CS-LBMFL outperforms the other multi-scale feature learning methods.

4) Comparisons With Existing Cost-Sensitive Learning Methods: We compared our cost-sensitive feature learning approach with two existing cost-sensitive learning methods: cost-sensitive linear discriminant analysis (CS-LDA) [40] and cost-sensitive feature selection (CS-FS) [43]. For CS-LDA, we used the linear-truncate cost function to compute the between-class variation. For CS-FS, we selected the top 1000 important features for feature representation. The OHRank [5] age estimator was used for age prediction. Table IV shows the MAEs of different cost-sensitive learning methods. As can be seen, our CS-LBFL and CS-LBMFL outperform the existing cost-sensitive learning methods.

5) Comparisons With Different Age Estimators: We investigated the performance of our CS-LBFL and CS-LBMFL when different age estimators are used. We compared OHRank with support vector regression (SVR) [24]. For SVR, we followed the same setting in [24]. Table V shows the MAEs of different age estimators. As can be seen, OHrank outperforms SVR. However, the difference is not very large.

6) Performance Analysis of Different Factors: We conducted experiments to analyze the performance of our CS-LBFL method when the cost information and binary features were employed individually. We created two baseline methods: LBFL and CS-LFL. For LBFL, the cost function was not employed in our CS-LBFL, which means that the cost matrix is set as an equal matrix where each element in this matrix is set as the same cost value. For CS-LFL, the sgn function was not used in (1), which means that the learned W only projects each NCDV into a real-valued feature vector. All other procedures of both LBFL and CS-LFL followed the same settings of those in CS-LBFL. Table VI shows the MAEs and CSs of different variations of our feature learning method on the FG-NET dataset. We see that both the cost-sensitive information and binary information contribute to the final age estimation performance of our feature learning method. Moreover, binary feature representation can improve the performance of our methods more than the cost-sensitive information. That is because binary information can extract binary feature descriptor which is robust to local variations in many real-world face images.

TABLE VI

MAEs (Years Old) and CS (%) Comparisons of CS-LBFL of Our CS-LBFL Method With Different Learning Strategies on the FG-NET Dataset

Method	MAE	CS ($\theta = 5$)
LBFL	4.55	73.7
CS-LFL	4.75	72.5
CS-LBFL	**4.43**	**74.6**

TABLE VII

Computational Time (Second) Comparison of Different Feature Learning Methods

Method	Time
DFD	0.60
LQP	0.10
RICA	0.35
CS-LBFL	0.06
CS-LBMFL	0.18

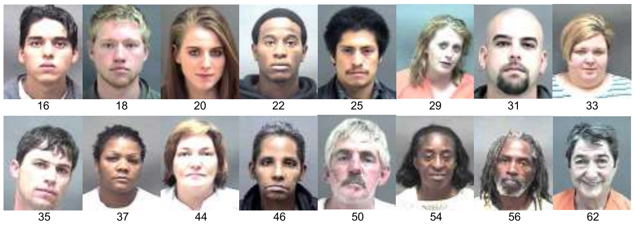


Fig. 10. Several example facial images with different age values in the MORPH (Album 2) dataset, where the number below each image is the age value of the person.

7) Computational Time: Lastly, we investigated the computational time of CS-LBFL and CS-LBMFL and compared them with other feature learning methods such as DFD, LQP, and RICA. All experiments were conducted on a PC with a 3.40GHz i7 CPU and a 24Gb RAM under the MATLAB platform. Table VII shows the computational time of different feature learning methods for one face image. We see that our CS-LBFL is faster and our CS-LBMFL are comparable with other feature learning methods.

C. Experiments on the MORPH Dataset

There are 55608 face images from more than 13000 subjects in the MORPH (Album 2) database [48]. The average number of face image per person is 4, and the age range of this dataset is from 16 to 77 years old. Fig. 10 shows some face images from the MORPH dataset. For each face image, we manually cropped and aligned it into 64×64 according to the eye positions. The color images were also converted into gray-scale ones before feature learning.

We applied the 10-fold cross validation strategy for performance evaluation on the MORPH (Album 2) dataset because there are more than 13000 subjects and it is very

TABLE VIII

MAEs (YEARS OLD) COMPARISON WITH THE STATE-OF-THE-ART FACIAL AGE ESTIMATION METHODS ON THE MORPH (ALBUM 2) DATASET

Method	MAE
KNN	9.64
SVM	7.34
AGES [16]	8.83
MTWGP [63]	6.28
RED-SVM [4]	6.49
OHRank [5]	5.69
IIS-LLD [15]	5.67
CPNN [15]	4.87
CA-SVR [6]	5.88
MFOR [56]	4.20
BIF+OLPP [20]	4.45
rKCCA [22]	3.98
rKCCA + SVM [22]	3.92
CS-LDA [40]	6.03
CS-FS [43]	6.59
CS-LBFL	**4.52**
CS-LBMFL	**4.37**

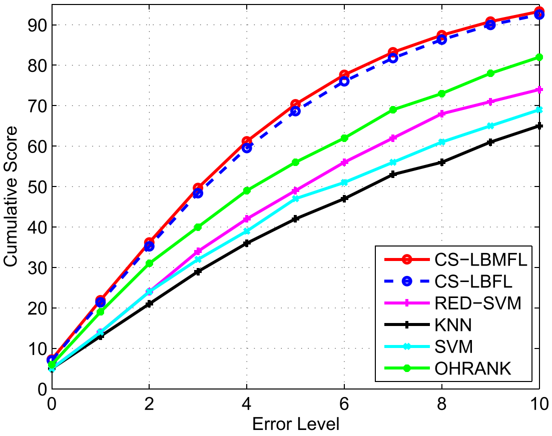


Fig. 11. CS curves comparisons of different facial age estimation methods on the MORPH (Album 2) dataset.

time-consuming to perform the LOPO test. Specifically, we divided the whole dataset into 10 folds and each fold has the nearly equal size. We used one fold as the testing set and the other nine folds for training. We repeated this procedure 10 times and computed the average result as the final estimation performance. We compared our methods with state-of-the-art facial age estimation methods and two existing cost-sensitive learning methods, as shown in Table VIII. Results of the existing state-of-the-art methods are obtained from the original papers and these two cost-sensitive learning methods are implemented by ourselves. The methods of BIF+OLPP [20] and rKCCA [22] exploited the information of human race and gender in their age estimation systems. Fig. 11 shows the CS curves of different age estimation methods. As can be seen, our CS-LBFL and CS-LBMFL achieve very competitive performance with existing state-of-the-art facial age estimation methods.

D. Experiments on the LifeSpan Dataset

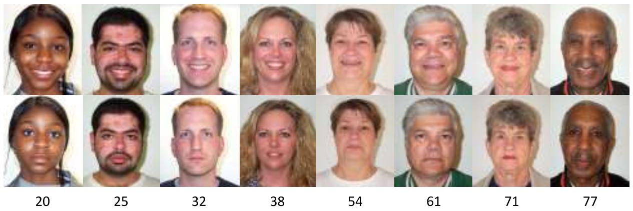


Fig. 12. Several example facial images with different age values in the LifeSpan dataset, Each column represents one person, where the top image shows a happy expression while the bottom image shows the neutral expression. The number below the image is the age value of the person.

TABLE IX

MAEs (YEARS OLD) OF DIFFERENT FACIAL AGE ESTIMATION METHODS ON THE LIFESPAN DATASET

Method	Neutral	Happy
BIF [25]	8.93	10.75
BIF+MFA [25]	6.05	7.36
CS-LDA [40]	8.18	9.35
CS-FS [43]	9.29	10.01
OHRank (SIFT) [5]	9.56	10.00
OHRank (LBP) [5]	8.63	8.69
CS-LBFL	**5.79**	**5.84**
CS-LBMFL	**5.26**	**5.62**

There are 844 face images in the LifeSpan dataset [44], and the age range is from 18 to 94 years old. Unlike the FG-NET and MORPH datasets, the LifeSpan dataset consists of face images of the same person captured from two different expressions. Specifically, there are 590 subjects in this dataset, and each subject has some face images with the neutral expression. Among these 590 subjects, some of them also contain the happy expression. Fig. 12 shows some face examples with different ages and expressions. For each face image, we manually cropped and aligned it into 128×128 according to the eye positions.

In our experiments, we performed age estimation for face images which were captured under the same expression. In other words, face images in both the training and testing sets have the same expression from different subjects. We performed the five-fold cross validation for each expression set and computed the MAE for comparison. We compared our methods with OHRank using two other hand-crafted features: LBP and SIFT. Specifically, we extracted LBP and SIFT histogram features from 8×8 non-overlapping blocks, and extract 59D LBP feature and 128D SIFT feature for each block, respectively. Finally, features from all blocks were concatenated together as the final feature representation, and OHRank was used for age prediction. We also compared our methods with CS-LDA and CS-FS, which are implemented by ourselves. Table IX shows the performance of different facial age estimation methods on the LifeSpan dataset. Fig. 13 shows the CS curve of different methods for the happy expression subset. As can be seen, our CS-LBFL and CS-LBMFL significantly improve the existing facial age estimation methods.

E. Experiments on the FACES Dataset

There are 2052 frontal face images of 171 subjects in the FACES dataset [9], and the age range of this dataset is from 19 to 80 years old. For each person, there are six expressions:

TABLE X

MAEs (YEARS OLD) OF DIFFERENT FACIAL AGE ESTIMATION METHODS ON THE FACES DATASET

Representation	Neutral	Happy	Disgust	Fearful	Sad	Angry
BIF [25]	9.50	10.70	13.26	12.65	10.78	13.26
BIF+MFA [25]	8.14	10.32	12.24	10.73	10.66	10.96
CS-LDA [40]	5.97	7.52	9.20	8.63	8.48	9.16
CS-FS [43]	7.83	8.78	8.85	9.44	9.16	10.99
OHRank (LBP) [5]	5.16	7.64	8.31	7.00	6.87	7.87
OHRank (SIFT) [5]	6.36	8.88	9.20	7.30	9.09	8.86
CS-LBFL	**5.06**	**6.53**	**7.15**	**6.32**	**6.27**	**6.94**
CS-LBMFL	**4.84**	**5.85**	**5.70**	**6.10**	**4.98**	**5.50**

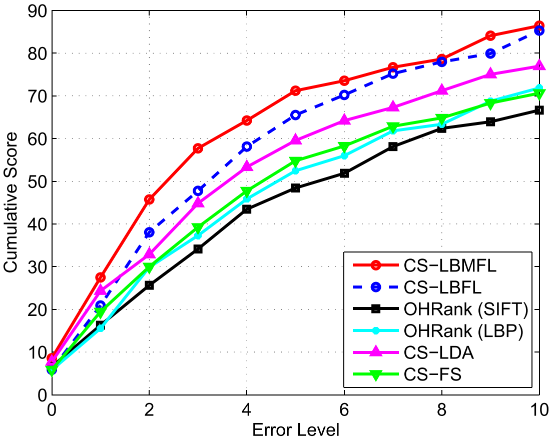


Fig. 13. CS curves of different facial age estimation methods on the LifeSpan dataset.

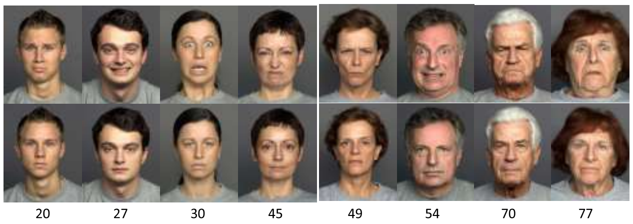


Fig. 14. Several example facial images with different age values in the FACES dataset, Each column represents one individual where the top image shows a specific expression (happy, sad, surprise, disgust, and fear) while the bottom image shows the neutral expression. The number below the image is the age value of the person.

neutral, sad, disgust, fear, angry and happy. Fig. 14 shows some face examples of three subjects with different ages and expressions in the FACES dataset. In our experiments, each face image was cropped and aligned to 128×128 according to the eye coordinates. We performed the five-fold cross validation and computed the age estimation performance under the same expression. We compared our methods with OHRank with LBP and SIFT features, where the procedure of extracting these two features is the same as that on the LifeSpan dataset. We also compared our methods with CS-LDA and CS-FS, which are implemented by ourselves. Table X shows the performance of our methods and the other baseline methods. Fig. 15 shows the CS curves of different methods on the sad expression subset of the FACES dataset. As can be seen, our CS-LBFL and CS-LBMFL methods significantly improve the existing state-of-the-art facial age estimation methods.

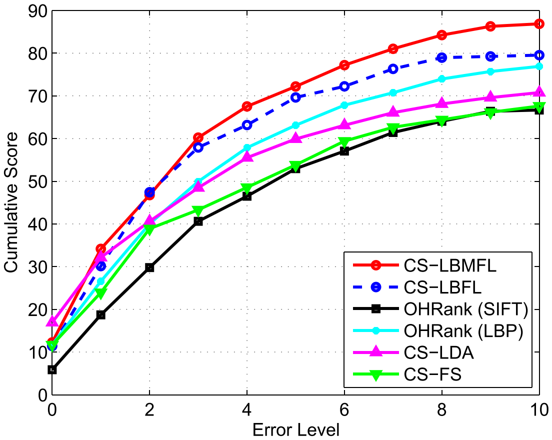


Fig. 15. CS curves of different facial age estimation methods on the sad expression subset of the FACES dataset.

F. Discussion

We make the following two observations from the above experimental results:

1) Our CS-LBFL and CS-LBMFL achieve very competitive performance with state-of-the-art facial age estimation in all the four datasets. This is because our CS-LBFL and CS-LBMFL automatically learn feature representation from raw data, which are more data-adaptive than the hand-crafted feature descriptors which are used in most existing methods. Moreover, both CS-LBFL and CS-LBMFL are binary feature descriptors, which demonstrate stronger robustness to local variations.
2) Both the cost-sensitive information and binary information contribute to the performance of our proposed methods, and binary information can improve the performance more than the cost-sensitive information. That is because binary information can extract binary feature descriptor which is robust to local variations in many real-world face images.

V. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a cost-sensitive local binary feature learning (CS-LBFL) method for facial age estimation. Since facial age estimation is a cost-sensitive computer vision problem and local binary features are robust to different variations, we learned a series of hashing functions to project raw pixel values into low-dimensional binary codes and encoded them into a real-value histogram feature for face representation. Moreover, we proposed a cost-sensitive binary multiple feature learning (CS-LBMFL) method by extracting multiple NCDVs to exploit complementary information to further improve the performance. Experimental results on four widely used face aging datasets show the efficacy of the proposed methods.

There are two interesting directions for future work:

1) Our CS-LBFL and CS-LBMFL are general feature learning methods and it is interesting to apply them to other visual recognition applications besides age estimation in this study.
2) In this work, we only learned features from one single layer and it is interesting to learn deep and hierarchal features for future study.

References

[1] T. Ahonen, A. Hadid, and M. Pietikäinen, "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] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Adv. NIPS*, 2007, pp. 153–160.

[3] K.-Y. Chang and C.-S. Chen, "A learning framework for age rank estimation based on face images with scattering transform," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 785–798, Mar. 2015.

[4] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "A ranking approach for human ages estimation based on face images," in *Proc. 20th ICPR*, Aug. 2010, pp. 3396–3399.

[5] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 585–592.

[6] K. Chen, S. Gong, T. Xiang, and C. C. Loy, "Cumulative attribute space for age and crowd density estimation," in *Proc. IEEE Conf. CVPR*, Jun. 2013, pp. 2467–2474.

[7] Y.-L. Chen and C.-T. Hsu, "Subspace learning for facial age estimation via pairwise age ranking," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 12, pp. 2164–2176, Dec. 2013.

[8] P. Domingos, "MetaCost: A general method for making classifiers cost-sensitive," in *Pros. 5th ACM SIGKDD Int. Conf. KDD*, 1999, pp. 155–164.

[9] N. C. Ebner, M. Riediger, and U. Lindenberger, "FACES—A database of facial expressions in young, middle-aged, and older women and men: Development and validation," *Behavior Res. Methods*, vol. 42, no. 1, pp. 351–362, 2010.

[10] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. 17th IJCAI*, 2001, pp. 973–978.

[11] Y. Fu, G. Guo, and T. S. Huang, "Age synthesis and estimation via faces: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1955–1976, Nov. 2010.

[12] Y. Fu and T. S. Huang, "Human age estimation with regression on discriminative aging manifold," *IEEE Trans. Multimedia*, vol. 10, no. 4, pp. 578–584, Jun. 2008.

[13] X. Geng and K. Smith-Miles, "Facial age estimation by multilinear subspace analysis," in *Proc. IEEE ICASSP*, Apr. 2009, pp. 865–868.

[14] X. Geng, K. Smith-Miles, and Z.-H. Zhou, "Facial age estimation by nonlinear aging pattern subspace," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 721–724.

[15] X. Geng, C. Yin, and Z.-H. Zhou, "Facial age estimation by learning from label distributions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 10, pp. 2401–2412, Oct. 2013.

[16] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, Dec. 2007.

[17] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *Proc. Conf. CVPR*, Jun. 2011, pp. 817–824.

[18] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1178–1188, Jul. 2008.

[19] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "A probabilistic fusion approach to human age prediction," in *Proc. IEEE Comput. Soc. Conf. CVPRW*, Jun. 2008, pp. 1–6.

[20] G. Guo and G. Mu, "Human age estimation: What is the influence across race and gender?" in *Proc. IEEE Comput. Soc. Conf. CVPRW*, Jun. 2010, pp. 71–78.

[21] G. Guo and G. Mu, "Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 657–664.

[22] G. Guo and G. Mu, "A framework for joint estimation of age, gender and ethnicity on a large database," *Image Vis. Comput.*, vol. 32, no. 10, pp. 761–770, 2014.

[23] G. Guo, G. Mu, Y. Fu, C. Dyer, and T. Huang, "A study on automatic age estimation using a large database," in *Proc. IEEE 12th ICCV*, Sep./Oct. 2009, pp. 1986–1991.

[24] G. Guo, G. Mu, Y. Fu, and T. S. Huang, "Human age estimation using bio-inspired features," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 112–119.

[25] G. Guo and X. Wang, "A study on human age estimation under facial expression changes," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2547–2553.

[26] A. Hadid, M. Pietikäinen, and T. Ahonen, "A discriminative feature space for detecting and recognizing faces," in *Proc. IEEE Comput. Soc. Conf. CVPR*, Jun./Jul. 2004, pp. II-797–II-804.

[27] H. Han, C. Otto, and A. K. Jain, "Age estimation from face images: Human vs. machine performance," in *Proc. ICB*, Jun. 2013, pp. 1–8.

[28] 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.

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

[30] S. ul Hussain, T. Napoléon, and F. Jurie, "Face recognition using local quantized patterns," in *Proc. BMVC*, 2012, pp. 1–11.

[31] J. Kittler, A. Hilton, M. Hamouz, and J. Illingworth, "3D assisted face recognition: A survey of 3D imaging, modelling and recognition approachest," in *Proc. IEEE Comput. Soc. Conf. CVPR Workshops*, Jun. 2004, p. 114.

[32] Y. H. Kwon and N. da Vitoria Lobo, "Age classification from facial images," *Comput. Vis. Image Understand.*, vol. 74, no. 1, pp. 1–21, 1999.

[33] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Toward automatic simulation of aging effects on face images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 442–455, Apr. 2002.

[34] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng, "ICA with reconstruction cost for efficient overcomplete feature learning," in *Proc. Adv. NIPS*, 2011, pp. 1017–1025.

[35] Z. Lei, M. Pietikäinen, and S. Z. Li, "Learning discriminant face descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 289–302, Feb. 2014.

[36] C. Li, Q. Liu, J. Liu, and H. Lu, "Learning ordinal discriminative features for age estimation," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2570–2577.

[37] Y. Li, J. T.-Y. Kwok, and Z.-H. Zhou, "Cost-sensitive semi-supervised support vector machine," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, pp. 500–505.

[38] 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.

[39] X.-Y. Liu and Z.-H. Zhou, "The influence of class imbalance on cost-sensitive learning: An empirical study," in *Proc. 6th ICDM*, Dec. 2006, pp. 970–974.

[40] J. Lu and Y.-P. Tan, "Cost-sensitive subspace learning for face recognition," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 2661–2666.

[41] J. Lu and Y.-P. Tan, "Cost-sensitive subspace analysis and extensions for face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 3, pp. 510–519, Mar. 2013.

[42] H. Masnadi-Shirazi and N. Vasconcelos, "Cost-sensitive boosting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 294–309, Feb. 2011.

[43] L. Miao, M. Liu, and D. Zhang, "Cost-sensitive feature selection with application in software defect prediction," in *Proc. 21st ICPR*, Nov. 2012, pp. 967–970.

[44] M. Minear and D. C. Park, "A lifespan database of adult facial stimuli," *Behavior Res. Methods, Instrum., Comput.*, vol. 36, no. 4, pp. 630–633, 2004.

[45] A. Montillo and H. Ling, "Age regression from faces using random forests," in *Proc. 16th IEEE ICIP*, Nov. 2009, pp. 2465–2468.

[46] T. Ojala, M. Pietikäinen, and T. Mäenpää, "A generalized local binary pattern operator for multiresolution gray scale and rotation invariant texture classification," in *Proc. 2nd ICAPR*, 2001, pp. 399–408.

[47] S. Raudys and A. Raudys, “Pairwise costs in multiclass perceptrons,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1324–1328, Jul. 2010.

[48] K. Ricanek, Jr., and T. Tesafaye, “MORPH: A longitudinal image database of normal adult age-progression,” in *Proc. 7th Int. Conf. FGR*, Apr. 2006, pp. 341–345.

[49] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, “Contractive auto-encoders: Explicit invariance during feature extraction,” in *Proc. 28th ICML*, 2011, pp. 833–840.

[50] C. Shan, S. Gong, and P. W. McOwan, “Facial expression recognition based on local binary patterns: A comprehensive study,” *Image Vis. Comput.*, vol. 27, no. 6, pp. 803–816, 2009.

[51] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, “Cost-sensitive boosting for classification of imbalanced data,” *Pattern Recognit.*, vol. 40, no. 12, pp. 3358–3378, 2007.

[52] Y. Sun, X. Wang, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in *Proc. IEEE Conf. CVPR*, Jun. 2014, pp. 1891–1898.

[53] K. M. Ting, “A comparative study of cost-sensitive boosting algorithms,” in *Proc. 17th ICML*, 2000, pp. 983–990.

[54] J. Wang, S. Kumar, and S.-F. Chang, “Semi-supervised hashing for scalable image retrieval,” in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 3424–3431.

[55] K. Q. Weinberger and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” *J. Mach. Learn. Res.*, vol. 10, pp. 207–244, Feb. 2009.

[56] R. Weng, J. Lu, G. Yang, and Y.-P. Tan, “Multi-feature ordinal ranking for facial age estimation,” in *Proc. 10th IEEE Int. Conf. Workshops FG*, Apr. 2013, pp. 1–6.

[57] B. Xiao, X. Yang, Y. Xu, and H. Zha, “Learning distance metric for regression by semidefinite programming with application to human age estimation,” in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 451–460.

[58] S. Xie, S. Shan, X. Chen, and J. Chen, “Fusing local patterns of Gabor magnitude and phase for face recognition,” *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1349–1361, May 2010.

[59] S. Yan, H. Wang, Y. Fu, J. Yan, X. Tang, and T. S. Huang, “Synchronized submanifold embedding for person-independent pose estimation and beyond,” *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 202–210, Jan. 2009.

[60] S. Yan, H. Wang, X. Tang, and T. S. Huang, “Learning auto-structured regressor from uncertain nonnegative labels,” in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.

[61] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: A general framework for dimensionality reduction,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

[62] B. Zadrozny, J. Langford, and N. Abe, “Cost-sensitive learning by cost-proportionate example weighting,” in *Proc. 3rd IEEE ICDM*, Nov. 2003, pp. 435–442.

[63] Y. Zhang and D.-Y. Yeung, “Multi-task warped Gaussian process for personalized age estimation,” in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 2622–2629.

[64] Z.-H. Zhou and X.-Y. Liu, “Training cost-sensitive neural networks with methods addressing the class imbalance problem,” *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 1, pp. 63–77, Jan. 2006.

Jiwen Lu (M’11–SM’15) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi’an University of Technology, Xi’an, China, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore. He is currently a Faculty Member with the Department of Automation, Tsinghua University, China. His research interests include computer vision, pattern recognition, and machine learning. He has authored/co-authored over 110 scientific papers in these areas. He serves as an Associate Editor of *Pattern Recognition Letters*, *Neurocomputing*, and the IEEE ACCESS. He was a recipient of the First-Prize National Scholarship and the National Outstanding Student Award from the Ministry of Education of China in 2002 and 2003, the Best Student Paper Award from the Pattern Recognition and Machine Intelligence Association of Singapore in 2012, the Top 10% Best Paper Award from the IEEE International Workshop on Multimedia Signal Processing in 2014, and the National 1000 Young Talents Plan Program in 2015, respectively.

Venice Erin Liong received the B.S. degree from the University of the Philippines Diliman, Quezon City, Philippines, in 2010, and the M.S. degree from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2013. She is currently pursuing the Ph.D. degree with the Interdisciplinary Graduate School, Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore. Her research interests include computer vision and pattern recognition.

Jie Zhou (M’01–SM’04) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 1995. Since 1997, he has served as a post-doctoral fellow with the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a Full Professor with the Department of Automation, Tsinghua University. His research interests include computer vision, pattern recognition, and image processing. He has authored over 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.