Neurocomputing xxx (xxxx) xxx

Contents lists available at ScienceDirect

Neurocomputing

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

Cost-sensitive joint feature and dictionary learning for face recognition

Guoing Zhang[a,*], Fatih Porikli[b], Huaijiang Sun[c], Quansen Sun[c], Guiyu Xia[a], Yuhui Zheng[a]

[a] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[b] Research School of Engineering, Australian National University, Canberra ACT 2601, Australia
[c] School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China

ARTICLE INFO

Article history:
Received 22 January 2018
Revised 6 December 2019
Accepted 27 January 2020
Available online xxx

Communicated by Jungong Han

Keywords:
Face recognition
Cost-sensitive learning
Feature learning
Dictionary learning
Joint learning

ABSTRACT

Dictionary learning (DL) for classification aims to learn a codebook from training samples to enhance the discriminative capability of their coding vectors. But how to determine appropriate features that can best work with the learned dictionary remains an open question. Recently, several joint feature and dictionary learning method have been proposed and achieved impressive performance. The purpose of these methods is to achieve low classification errors by implicitly assuming the costs for all misclassifications, regardless of the original labels, are the same. However, in real applications, this assumption may not hold as different kinds of mistake could produce different losses. Motivated by this concern, we propose a cost-sensitive joint feature and dictionary learning (CS-JFDL) method, in which the features are concurrently learned with the dictionaries. Our method considers the separate misclassification cost objectives during the feature and dictionary learning stages to achieve a minimum overall recognition loss. Thus, the derived feature and dictionary attain cost-sensitive constraints throughout the learning process. Extensive experimental results on both image based face recognition and image set based face recognition demonstrate that the proposed algorithm is better than many state-of-the-art methods.

1. Introduction

Face recognition is an important research topic in computer vision and pattern recognition community over the past two decades. Up to now, there have been a number of face recognition methods have been proposed and successfully applied [1–9]. Despite achieved great progress, there are still many challenging in face recognition scenarios, when face samples are captured in unconstrained environments such as varying poses, illuminations expressions and resolutions. In such situations, the recognition performance of many methods will be heavily affected and significantly degraded. Hence, it is necessary to learn robust and discriminative feature representation before perform face identification.

Recently, dictionary-based methods have attracted increasing interest and achieved competitive performance on face recognition [10–20,59], because of its robust discriminant representation for the variations of expression, illumination and pose information within face samples that can be implicitly encoded into the learned dictionaries. Due to face images usually lie on a low-dimensional manifold, it is necessary to find the most discriminative representations in a low dimensional space, researchers have developed a series of joint feature and dictionary learning methods and reported more competitive performance [11–14,20,21]. However, most of these dictionary-based face recognition methods attempt to pursuit a minimum recognition rate and implicitly assume that the costs for all misclassification errors, regardless of the original labels, are the same. Although this assumption is reasonable, however, it is not practical, since different types of errors may cause different amounts of losses. For instance, in an access control system, it may be inconvenient to misclassify a permitted (gallery) person as an impostor and deny access, but it may lead to a *serious* security breach if an impostor is misrecognized as a gallery person and authorized to access. Thus, the false acceptance (misrecognizing an impostor as a gallery subject) may not be as serious as the false rejection (misrecognizing a gallery subject as an impostor) or even the false identification (misrecognizing between two gallery subjects). We can see that the three kinds of errors mentioned above are different and it is not a good choice to adopt error rate as the final measure criterion.

Inspired by the above considerations, we propose a robust cost-sensitive joint feature and dictionary learning method (CS-JFDL), in which the discriminative projection matrix is simultaneously learned with the structured dictionary. The basic idea is shown in Fig. 1. We show that the jointly learned discriminative features and class-specific dictionaries are complementary each other, thus

* Corresponding author.
E-mail addresses: xiayang14551@163.com (G. Zhang), zhengyh@vip.126.com (Y. Zheng).

https://doi.org/10.1016/j.neucom.2020.01.101
0925-2312/

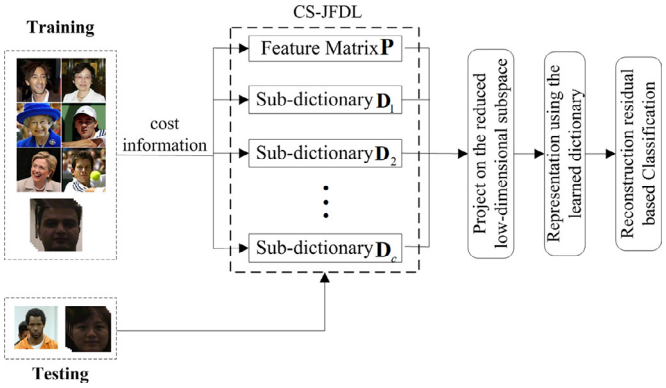

Fig. 1. Overview of the proposed CS-JFDL method.

more discriminative information for classification can be obtained. To achieve a minimum overall recognition cost, our method considers the cost information during the feature learning and dictionary learning stages. This makes the features and dictionary we learn cost-sensitive, which further improves the recognition performance when they are combined. Specifically, the cost-sensitive dictionary can produce cost-sensitive sparse coding while encouraging the samples from the same class to have similar sparse codes and those from different classes to have dissimilar sparse codes.

The rest of paper is organized as follows. Section 2 gives the related works. Our algorithm is described in Section 3, and the optimization procedure of CS-JFDL is explained in Section 4. Section 5 presents our classification scheme. Comparative experiments analysis is provided in Section 6. Section 7 gives the conclusion.

2. Related work

2.1. Feature learning

Learning useful and computationally convenient feature representation from complex, redundant, and highly variable visual data is essential in many computer vision tasks such as pedestrian detection [22], image classification [23,31], action recognition [24], and visual tracking [25]. Many feature learning methods have been proposed in recent years [24–26,60,61]. Recently, feature learning has also been exploited for face recognition and a lot of feature learning-based face recognition approaches have been developed [7,27–29]. Cao et al. [27] presented a learning-based feature representation method by applying the bag-of-word framework. Lu et al. [6] proposed a compact binary face descriptor feature learning method for face representation and recognition. Since face images are sensitive to the variations of illumination, occlusion and posed, it is desirable to learn robust and discriminative features for face image.

2.2. Dictionary learning

Dictionary learning has recently made significant improvement to a variety of recognition tasks for its excellent representation power [16–19,30]. Given a set of training samples $\boldsymbol{Y} = [\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_c] = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n] \in R^{m \times n}$ from c different subjects, the basic model of dictionary learning is typically posed as the minimization of the following optimization problem:

$$\|\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_F^2 + \lambda\|\boldsymbol{X}\|_1 \quad s.t. \|\boldsymbol{d}_j\|_2 = 1, \quad \forall j \tag{1}$$

where $\|\boldsymbol{Y}\|_F$ denotes the Frobenius norm defined as $\|\boldsymbol{Y}\|_F = \sqrt{\sum_{i,j}\|Y_{i,j}\|^2}$, $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_K] \in R^{m \times K}$ is the sought dictionary, K is the number of atoms in the learned dictionary, $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in R^{K \times n}$ is the sparse representation matrix of input samples $\boldsymbol{Y}$, λ is a scalar constant. Each dictionary item $\boldsymbol{d}_j$ is l_2 normalized. In this case, Eq. (1) can solved by using the algorithm provided in [41].

2.3. Cost-sensitive learning

Cost-sensitive learning is an interest topic in computer vision and pattern recognition areas [32–40]. In such settings, the "cost" information of different samples is introduced to characterize their importance to reflect different amounts of losses. It aims to minimize *total cost* rather than *total error*. Face recognition is generally a cost-sensitive learning problem and many successful cost-sensitive face recognition algorithms have been developed [35–40]. Zhang et al. [38] formulated face recognition problem as a multiclass cost-sensitive learning task and proposed two cost-sensitive classification methods. By using the cost information, Lu et al. [36,37] proposed four cost-sensitive discriminative subspace learning algorithms. Zhang et al. [40] presented a cost-sensitive dictionary learning algorithm for face recognition, while ignoring the contribution of features during the learning process.

3. Proposed methods

The purpose of CS-JFDL is to simultaneous learn a cost-sensitive discriminative projection matrix and a cost-sensitive dictionary to project each testing sample into a discriminative space, and then encode the sample over the learned dictionary.

In order to exploit the "cost" information of different samples during the dictionary learning process, we formulate the CS-JFDL model as

$$\min_{\boldsymbol{D},\boldsymbol{P},\boldsymbol{X}} J = R(\boldsymbol{D},\boldsymbol{P},\boldsymbol{X}) + \lambda_1 G(\boldsymbol{P}) + \lambda_2 H(\boldsymbol{X}) + \lambda_3\|\boldsymbol{X}\|_1 \quad s.t. \boldsymbol{P}\boldsymbol{P}^T = \boldsymbol{I}. \tag{2}$$

where $\boldsymbol{D}$ is the learned discriminative dictionary, $\boldsymbol{P} \in R^{d \times m}$ is the cost-sensitive feature projection matrix, d is the dimension of the learned feature space, $R(\boldsymbol{D}, \boldsymbol{P}, \boldsymbol{X})$ is the reconstruction error, $\|\boldsymbol{X}\|_1$ is the sparsity penalty, $G(\boldsymbol{P})$ is the cost-sensitive discriminative projection term, $H(\boldsymbol{X})$ is the cost-sensitive term imposed on the sparse representation coefficient matrix, λ_1, λ_2, λ_3 are mixing parameters to balance the importance of different terms.

3.1. Cost-sensitive discriminative projection term

We expect that the learned projection contains discriminative and cost-sensitive information. Therefore, to increase the discriminative power of projection $\boldsymbol{P}$, we constrain the samples from the same class to be similar and the samples from different classes to be significantly dissimilar, such that discriminative information can be discovered when learning projection matrix. We construct an intrinsic graph and a penalty graph to describe the geometrical information of the training data. The weight of the intrinsic graph is defined by

$$\boldsymbol{W}_{ij}^1 = \begin{cases} 1, & \text{if} \quad i \in S_{k_1}^+(j) \quad \text{or} \quad j \in S_{k_1}^+(i) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $S_{k_1}^+(i)$ represents the index set of the k_1 nearest neighbors of $\boldsymbol{y}_i$ in the same class.

Similarly, the weight of the penalty graph is defined by

$$\boldsymbol{W}_{ij}^2 = \begin{cases} 1, & \text{if} \quad j \in S_{k_2}^-(i) \quad \text{or} \quad i \in S_{k_2}^-(j) \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $S_{k_2}^-(i)$ represents the index set of the k_2 nearest neighbors of $\boldsymbol{y}_i$ from the other classes (not the class that $\boldsymbol{y}_i$ belongs to). Here, graphs $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$ are used to characterize the intra-class

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

compactness and inter-class separability of the training data, respectively.

In order to make the features cost-sensitive, we build a cost matrix $\boldsymbol{C}$, where $\boldsymbol{C}_{p,q}$ denotes the cost value of classifying the pth class sample as the qth class. The diagonal elements in $\boldsymbol{C}$ are zeros because there is no loss for a correct classification. In our paper, the cost matrix is assumed to be specified by the user. Therefore, we pay attention to how to learn cost-sensitive feature and dictionary to increase the classification accuracy. To maximize both the intra-class compactness ad inter-class separability, we formulate the cost-sensitive discriminative projection term with the following equivalent

$$\begin{aligned} G(\boldsymbol{P}) &= G_1(\boldsymbol{P}) - G_2(\boldsymbol{P}) \\ &= \sum_{i,j=1}^{N} \sum_{i\in S_{k_1}^{+}(j) \text{ or } j\in S_{k_1}^{+}(i)} \frac{1}{2} cost(\boldsymbol{y}_i, \boldsymbol{y}_j) \|\boldsymbol{P}\boldsymbol{y}_i - \boldsymbol{P}\boldsymbol{y}_j\|_2^2 \\ &\quad - \sum_{i,j=1}^{N} \sum_{i\in S_{k_2}^{-}(j) \text{ or } j\in S_{k_2}^{-}(i)} \frac{1}{2} cost(\boldsymbol{y}_i, \boldsymbol{y}_j) \|\boldsymbol{P}\boldsymbol{y}_i - \boldsymbol{P}\boldsymbol{y}_j\|_2^2 \end{aligned} \tag{5}$$

where $cost(\boldsymbol{y}_i, \boldsymbol{y}_j) = \boldsymbol{C}_{\ell_{\boldsymbol{y}_i}, \ell_{\boldsymbol{y}_j}}$, and $\ell_{\boldsymbol{y}_i}$ is the label of $\boldsymbol{y}_i$. Denoting $\boldsymbol{E}_{i,j} = \boldsymbol{C}_{\ell_{\boldsymbol{y}_i}, \ell_{\boldsymbol{y}_j}}$, the $G_2(\boldsymbol{P})$ can be expressed as follows:

$$\begin{aligned} G_2(\boldsymbol{P}) &= tr\left(\frac{1}{2}\sum_{i,j=1}^{N} (\boldsymbol{P}\boldsymbol{y}_i - \boldsymbol{P}\boldsymbol{y}_j)^2 \boldsymbol{E}_{ij}^2 \boldsymbol{W}_{ij}^2\right) \\ &= tr\left(\sum_i \boldsymbol{P}\boldsymbol{y}_i \boldsymbol{B}_{ii}^2 \boldsymbol{F}_{ii}^2 \boldsymbol{y}_i^T \boldsymbol{P}^T - \sum_{i,j} \boldsymbol{P}\boldsymbol{y}_i \boldsymbol{E}_{ij}^2 \boldsymbol{W}_{ij}^2 \boldsymbol{y}_j^T \boldsymbol{P}^T\right) \\ &= tr\left(\boldsymbol{P}\boldsymbol{Y}\boldsymbol{B}^2 \odot \boldsymbol{F}^2 \boldsymbol{Y}^T \boldsymbol{P}^T - \boldsymbol{P}\boldsymbol{Y}\boldsymbol{E}^2 \odot \boldsymbol{W}^2 \boldsymbol{Y}^T \boldsymbol{P}^T\right) \\ &= tr\left(\boldsymbol{P}\boldsymbol{Y}\boldsymbol{L}^2 \boldsymbol{Y}^T \boldsymbol{P}^T\right) \end{aligned} \tag{6}$$

where $\boldsymbol{L}^2 = \boldsymbol{B}^2 \odot \boldsymbol{F}^2 - \boldsymbol{E}^2 \odot \boldsymbol{W}^2$, $\odot$ denotes the element-wise multiplication, and $\boldsymbol{B}^2$, $\boldsymbol{F}^2$ are diagonal matrices, whose entries are column (or row, since $\boldsymbol{E}^2$ and $\boldsymbol{W}^2$ are symmetric) sums of $\boldsymbol{E}^2$ and $\boldsymbol{W}^2$, in which $\boldsymbol{B}^2 = \sum_{j\neq i} \boldsymbol{E}_{ij}^2$, $\boldsymbol{F}^2 = \sum_{j\neq i} \boldsymbol{W}_{ij}^2$.

3.2. Cost-sensitive sparse coding term

For obtaining the cost-sensitive sparse codes with the learned dictionary, we formulate the third term in Eq. (2) as follows

$$H(\boldsymbol{X}) = \|\boldsymbol{Q} \odot \boldsymbol{X}\|_F^2 \tag{7}$$

where $\boldsymbol{Q} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_n] \in R^{K\times n}$ is the penalizing matrix of input samples $\boldsymbol{Y}$, and K is the number of dictionary atoms. We call $\boldsymbol{q}_i$ is the cost-sensitive adaptor whose kth element is given by $q_i^k = \delta(C_{\ell_{\boldsymbol{y}_i}, \ell_{d_k}})$, $i = 1, \ldots, K$, where $C_{\ell_{\boldsymbol{y}_i}, \ell_{d_k}}$ indicates the cost of misclassifying the sample of class $\ell_{\boldsymbol{y}_i}$ as class ℓ_{d_k}, and $\delta(\cdot)$ is a discrete impulse function and defined as

$$\delta(\rho) = \begin{cases} 1, & \rho = 0 \\ \sigma\rho, & \rho \neq 0 \end{cases} \tag{8}$$

where σ is a cost constant. For example, assuming $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_6]$ and $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_6]$, where $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, $\boldsymbol{d}_1$, $\boldsymbol{d}_2$ are from class 1, $\boldsymbol{y}_3$, $\boldsymbol{y}_4$, $\boldsymbol{d}_3$, $\boldsymbol{d}_4$ are from class 2, and $\boldsymbol{y}_5$, $\boldsymbol{y}_6$, $\boldsymbol{d}_5$, $\boldsymbol{d}_6$ are from class 3, $\boldsymbol{Q}$ can be expressed as follows:

$$Q \equiv \begin{bmatrix} \delta(\boldsymbol{C}_{1,1}) & \delta(\boldsymbol{C}_{1,1}) & \delta(\boldsymbol{C}_{2,1}) & \delta(\boldsymbol{C}_{2,1}) & \delta(\boldsymbol{C}_{3,1}) & \delta(\boldsymbol{C}_{3,1}) \\ \delta(\boldsymbol{C}_{1,1}) & \delta(\boldsymbol{C}_{1,1}) & \delta(\boldsymbol{C}_{2,1}) & \delta(\boldsymbol{C}_{2,1}) & \delta(\boldsymbol{C}_{3,1}) & \delta(\boldsymbol{C}_{3,1}) \\ \delta(\boldsymbol{C}_{1,2}) & \delta(\boldsymbol{C}_{1,2}) & \delta(\boldsymbol{C}_{2,2}) & \delta(\boldsymbol{C}_{2,2}) & \delta(\boldsymbol{C}_{3,2}) & \delta(\boldsymbol{C}_{3,2}) \\ \delta(\boldsymbol{C}_{1,2}) & \delta(\boldsymbol{C}_{1,2}) & \delta(\boldsymbol{C}_{2,2}) & \delta(\boldsymbol{C}_{2,2}) & \delta(\boldsymbol{C}_{3,2}) & \delta(\boldsymbol{C}_{3,2}) \\ \delta(\boldsymbol{C}_{1,3}) & \delta(\boldsymbol{C}_{1,3}) & \delta(\boldsymbol{C}_{2,3}) & \delta(\boldsymbol{C}_{2,3}) & \delta(\boldsymbol{C}_{3,3}) & \delta(\boldsymbol{C}_{3,3}) \\ \delta(\boldsymbol{C}_{1,3}) & \delta(\boldsymbol{C}_{1,3}) & \delta(\boldsymbol{C}_{2,3}) & \delta(\boldsymbol{C}_{2,3}) & \delta(\boldsymbol{C}_{3,3}) & \delta(\boldsymbol{C}_{3,3}) \end{bmatrix} \tag{9}$$

where row represents the dictionary atoms and column represents the training samples. Each column is the cost penalty coefficient for an input image.

The "cost" penalizing matrix $\boldsymbol{Q}$ can make the sparse codes to have cost-sensitivity in sparse feature space. It aims to encourage the sample from the same class have very similar sparse codes (i.e., encouraging cost-sensitive in the results codes). This regularization term would penalize the non-zero entries whose corresponding atoms have different labels with the input samples.

3.3. Discriminative reconstruction error term

To further enhance the discrimination of the learned dictionary, in the reduced subspace, we not only require the whole dictionary $\boldsymbol{D}$ can properly reconstruct the input samples $\boldsymbol{Y}$, but also require the sub-dictionary $\boldsymbol{D}_i$ can well reconstruct samples from class i, and other sub-dictionary $\boldsymbol{D}_j$, $j \neq i$ is ineffectively to reconstruct samples from class i. We rewrite $\boldsymbol{X}_i$ as $\boldsymbol{X}_i = [\boldsymbol{X}_i^1; \ldots; \boldsymbol{X}_i^j; \ldots; \boldsymbol{X}_i^c]$, where $\boldsymbol{X}_i^j$ is the representation coefficients of $\boldsymbol{Y}_i$ over $\boldsymbol{D}_j$, $\boldsymbol{D}_j$ is the sub-dictionary of the jth class in $\boldsymbol{D}$. Thus, the reconstruction error term can be redefined as

$$R(\boldsymbol{D}, \boldsymbol{P}, \boldsymbol{X}) = \|\boldsymbol{P}\boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X}\|_F^2 + \sum_{i=1}^{c} \|\boldsymbol{P}\boldsymbol{Y}_i - \boldsymbol{D}_i \boldsymbol{X}_i^i\|_F^2 + \sum_{i=1}^{c} \sum_{j=1, j\neq i}^{c} \|\boldsymbol{D}_j \boldsymbol{X}_i^j\|_F^2. \tag{10}$$

where $\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots, \boldsymbol{D}_c]$ is the structured dictionary, $\boldsymbol{Y}_i$ denotes training samples of class i.

By incorporating Eq. (5), Eq. (7) and Eq. (10) into Eq. (2), the CS-JFDL model is formulated as:

$$\begin{aligned} &\min_{\boldsymbol{D},\boldsymbol{P},\boldsymbol{X}} J = R(\boldsymbol{D}, \boldsymbol{P}, \boldsymbol{X}) - \lambda_1 tr\left(\boldsymbol{P}\boldsymbol{Y}\boldsymbol{L}^2\boldsymbol{Y}^T\boldsymbol{P}^T\right) + \lambda_2 \|\boldsymbol{Q} \odot \boldsymbol{X}\|_F^2 + \lambda_3 \|\boldsymbol{X}\|_1, \\ &s.t. \boldsymbol{P}\boldsymbol{P}^T = \boldsymbol{I}. \end{aligned} \tag{11}$$

4. Optimization

The objective function in Eq. (11) is not convex for $\boldsymbol{D}$, $\boldsymbol{P}$, and $\boldsymbol{X}$ simultaneously, it is convex to one of them when the other two are fixed. We adopt an iterative learning framework to jointly learning the cost sensitive projection $\boldsymbol{P}$, the sparse representation $\boldsymbol{X}$ and cost sensitive dictionary $\boldsymbol{D}$. The complete optimization procedure is presented in Algorithm 1 and the detailed optimization process is provided in Appendix. Since the optimization model is non-convex, it is not guaranteed to converge to the global minimum. Fig. 2

Algorithm 1 Algorithm of CS-JFDL.

Input: training data $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n]$, Intrinsic graph $\boldsymbol{W}_{ij}^1$, penalty graph $\boldsymbol{W}_{ij}^2$, parameters λ_1, λ_2, λ_3 and σ, iteration number T.

Output: Cost-sensitive discriminative projection matrix $\boldsymbol{P}$, cost-sensitive dictionary $\boldsymbol{D}$, coding coefficient matrix $\boldsymbol{X}$.

Step 1: Initialization

Randomly initialize each column in $\boldsymbol{D}^0$ with unit l_2 norm. Initialize each column $\boldsymbol{x}_s$, $1 \le i \le s$ as $\left((\boldsymbol{D}^0)^T(\boldsymbol{D}^0) + \lambda\boldsymbol{I}\right)^{-1} \left(\boldsymbol{D}^0\right)^T \boldsymbol{y}_s$, where $\boldsymbol{y}_s$ is the s-th training samples (regardless of label).

Step 2: Local optimization

For $t = 1, 2, \ldots, T$, repeat

Solve $\boldsymbol{P}^t$ iteratively with fixed $\boldsymbol{D}^t$, $\boldsymbol{X}^{t-1}$ via Eq. (A.1).

Solve $\boldsymbol{X}^t$ with fixed $\boldsymbol{P}^t$, $\boldsymbol{D}^{t-1}$ via Eq. (A.9).

Solve $\boldsymbol{D}^t$ with fixed $\boldsymbol{P}^t$, $\boldsymbol{X}^t$ via Eq. (A.11).

Step 3: Output

Output $\boldsymbol{P} = \boldsymbol{P}^t, \boldsymbol{D} = \boldsymbol{D}^t, \boldsymbol{X} = \boldsymbol{X}^t$.

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

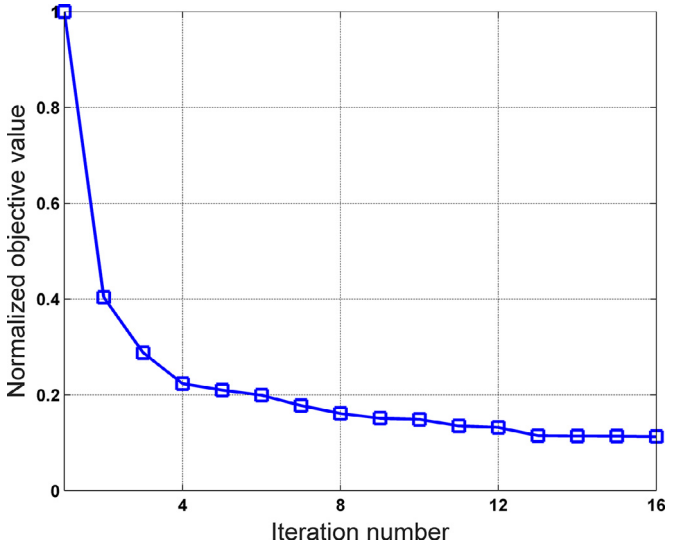

Fig. 2. The convergence of CS-JFDL on the AR dataset.

shows an example to illustrate the convergence behaviour of CS-JFDL. It seems that the objective function of our method can obtain stable performance in a few iterations.

The computational complexity of CS-JFDL comes from three parts: projection updating, sparse coding and dictionary learning. Suppose that the training set $\boldsymbol{Y}$ contains N samples with m dimension and dictionary $\boldsymbol{D}$ contains k atoms, the feature dimension is m. $\boldsymbol{P} \in R^{d \times m}$ is the feature projection matrix, where d is the dimension of low-dimension space. In the proposed algorithm, we use the feature-sign search algorithm [43] for learning coding coefficients with the ℓ_1 sparsity function, so the complexity of update coding coefficients for each sample is approximately $O(mk^2 + k^3)$. So the total time complexity of updating coding coefficients is $NO(mk^2 + k^3)$. The time complexity of updating dictionary atoms is $\Sigma_1 k_i O(2mN)$, where k_i is the number of dictionary atoms in $\boldsymbol{D}_i$. For projection updating, the time complexity is approximately $O(m^3)$. Therefore, the overall time complexity of CS-JFDL is approximately $t(NO(mk^2 + k^3)+) \sum_1 k_i O(2mN) + O(m^3)$, where t is the total number of iterations.

5. Classification strategy

When we get $\boldsymbol{P}$ and $\boldsymbol{D}$, coding vector for each projected sample can be obtained by solving the following formulation

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \|\boldsymbol{Py} - \boldsymbol{Dx}\|_2^2 + \alpha_1 \|\boldsymbol{x}\|_1 \right\} \tag{12}$$

where α_1 is a constant. Once $\hat{\boldsymbol{x}}$ is obtained, the reconstruction residual for each class is calculated by

$$r_i = \|\boldsymbol{Py} - \boldsymbol{D}_i \bar{\delta}_i(\hat{\boldsymbol{x}})\|_2^2, \quad for \quad i = 1, \ldots, c \tag{13}$$

where $\bar{\delta}_i(\cdot)$ is the characteristic function which chooses the coefficients corresponding to the ith class. The decision rule is: if $r_j(\boldsymbol{y}) = min_i r_i(\boldsymbol{y})$, $\boldsymbol{y}$ is assigned to class j.

Given a query sample of gallery class 3 from AR dataset [44]. Fig. 3(a) displays the coding coefficients of the query image over dictionary $\boldsymbol{D}$. From Fig. 3(a), we can see that the obtained absolute value of the coding coefficients of the query image have cost-sensitivity, i.e., the coding coefficients of small cost subjects (gallery) are significantly greater than the large cost subjects (impostor). The reconstruction residuals of the query image on each class are displayed in Fig. 3(b). We can observe that our approach gets the smallest residual in class 3 and the residuals of other gallery classes are smaller than impostor classes. i.e., even if the sample is misclassified, the query image will also be classified to other gallery classes for achieving a minimum loss.

We randomly select another query sample from the impostor classes. The sparse codes of the query sample are displayed in Fig. 4(a). It can be seen that the query sample can be well represented as a sparse liner combination of the atoms from impostor classes, and the coding coefficients from gallery classes near zero. Fig. 4(b) presents the reconstruction residuals. Note that, the residuals in impostor subjects are greater smaller than those of gallery classes. This implies that even if the query image is classified incorrectly, it can still be classified to other impostor subjects with greater probability, and then result in a lower cost.

6. Experiments

We evaluate our proposed algorithm CS-JFDL by using two typical applications including image based face recognition and image set based face recognition. In the first setup, we adopt four widely applied face datasets including AR [44], FERET [45], LFWa [46] and FRGC [47]. For image set based face recognition task. Three video face recognition benchmark datasets, including Honda/UCSD [48], CMU Mobo [49] and YouTube Celebrities (YTC) [50] are used to evaluate the proposed CS-JFDL.

6.1. Image based face recognition

The AR dataset consists of 4000 color images of 126 people (70 male and 56 female), which includes different lighting conditions, expressions, and facial disguise. The frontal view images without occlusion are used in our experiments, and each image is cropped and resized into 64 × 64 pixels.

The FERET dataset includes 1199 subjects with a total of 14,051 images, captured under various lighting, facial expressions, and pose. We select a subset of the FERET dataset which contain 200 individuals (each one has seven images), and only involves frontal view with different expressions and illumination for our experiments. Each image is cropped and resized to the size of 64 × 64 pixels.

The LFWa [46] dataset is an aligned version of LFW [51], which including different expression, illumination, pose misalignment and occlusion. We choose 143 subject with no less than 11 samples per subject (4174 images in total) to perform the experiment.

The FRGC dataset contains 12,776 training images, 16,028 controlled target images and 8014 uncontrolled query images, including 222 individuals, each 36–64 images. The controlled images have good image quality, while the uncontrolled images display poor image quality. We choose 36 images of each subject and crop each image to the size of 60 × 60 pixels.

6.1.1. Experimental settings

Let $\boldsymbol{C}_{GI}$, $\boldsymbol{C}_{IG}$ and $\boldsymbol{C}_{GG}$ be the different costs caused by a false rejection, a false acceptance and a false identification, respectively. For convenience of our discussion, we use $\boldsymbol{C}_{GI} = (\boldsymbol{C}_{GI}/\boldsymbol{C}_{GG})$, $\boldsymbol{C}_{IG} = (\boldsymbol{C}_{IG}/\boldsymbol{C}_{GG})$ and $\boldsymbol{C}_{GG} = 1$. This setting will not influence the experiment results.

The training and the testing set each contains N_G samples with M gallery classes and N_I samples with L impostor classes randomly selected from the entire dataset. The experiments are run10 times for each dataset and take the average result as the final recognition rate. Parameters M, N_G, N_I, L, $\boldsymbol{C}_{GI}$, $\boldsymbol{C}_{IG}$ and $\boldsymbol{C}_{GG}$ are specified in Table 1.

In all of our experiments, we set parameter $\sigma = 2$ and it always works well. The tuning parameters λ_1, λ_2 and λ_3 are evaluated by five-fold cross validation on the training data. There are chosen from {0.0001, 0.001, 0.005, 0.01, 0.1, 1, 2, 5}. Since there

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

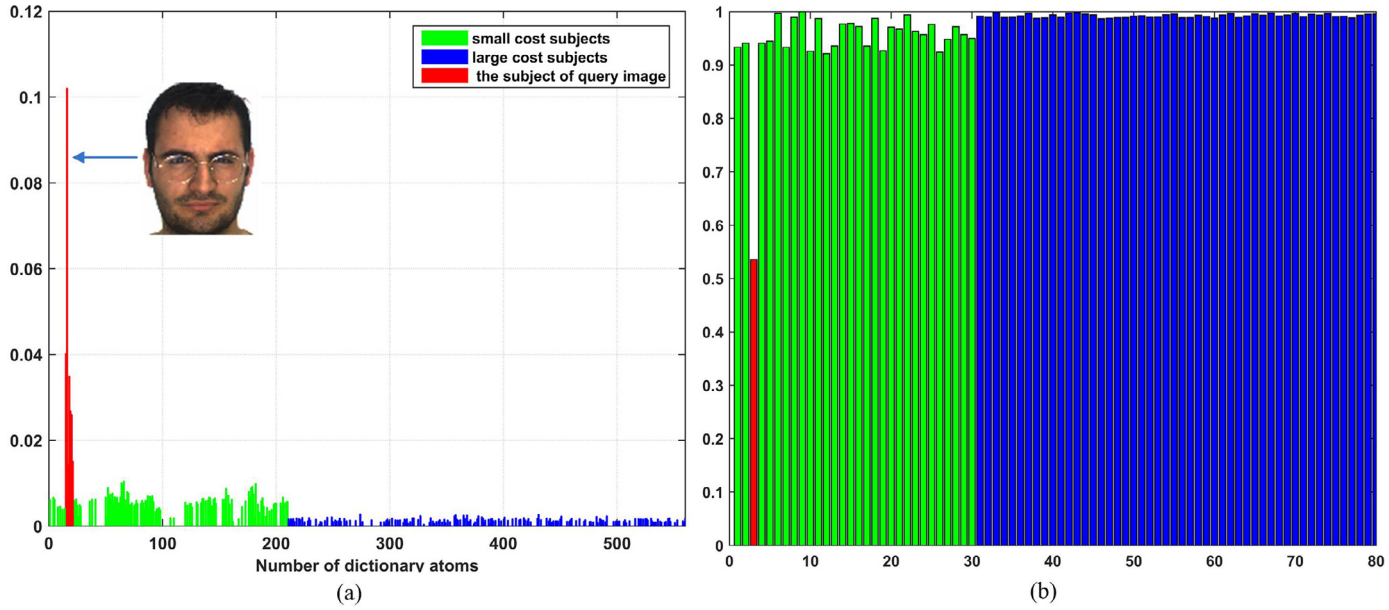

Fig. 3. (a) The sparse codes of the query image from gallery class 3. (b) The reconstruction residuals on each class.

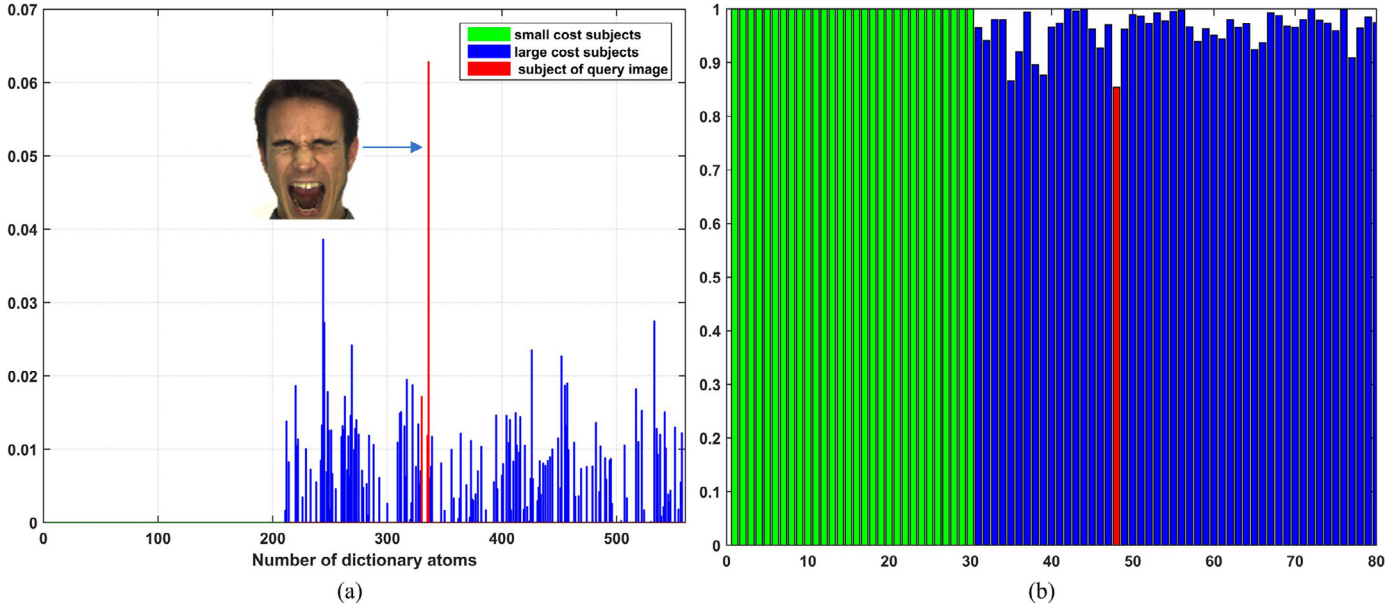

Fig. 4. (a) The sparse codes of the query image from imposter class. (b) The reconstruction residuals on each class.

Table 1
Experiments settings.

Datasets	M	N_G	N_I	L	C_{IG}: C_{GI}: C_{GG}
AR	30	7	7	50	20:2:1
FERET	70	3	3	110	20:2:1
LFWa	70	5	5	70	20:2:1
FRGC	40	15	15	30	20:2:1

are multiple parameters to be tuned and it is generally difficult to find them simultaneously. Consequently, we first fix parameters λ_2 and λ_3, and test the recognition performance of our method, then select the appropriate value for parameter λ_1. Similarly, we fix parameters λ_2, λ_3 and seek the optimal value for λ_2. Parameter λ_3 is determined using the same way. For AR [44], FERET [45] and FRGC [47] datasets, we empirically set $\lambda_1 = 1, \lambda_2 = 1$ and $\lambda_3 = 0.005$, and set $\lambda_1 = 1, \lambda_2 = 1$ and $\lambda_3 = 0.001$ on LFWa dataset [46]. The number of dictionary atoms in CS-JFDL is set as the number of training samples for each class. We construct intrinsic graph $\boldsymbol{W}_{ij}^1$ and penalty graph $\boldsymbol{W}_{ij}^2$ with correlation similarity. The Euclidean distances among the training samples are used as initial neighbor metric. We set $k_1 = 5, k_2 = 20$. Specifically, we use $\alpha_1 = 0.001$ for classification. For all the baseline approaches, we usually use their original settings, and for those that do not have open source, we carefully implement them following the paper.

6.1.2. Results and analysis

Comparisons with SoA Cost-Blind Methods: We compare our approach with some state-of-the-art dictionary learning algorithms, including Discriminative KSVD (D-KSVD) [10], Label Consistent K-SVD (LC-KSVD) [16], Fisher discrimination dictionary learning (FDDL) [18], Latent dictionary learning (LDL) [17], Dictionary learn-

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

Table 2
Comparison of cost-blind methods on AR and FERET datasets (*cost*: total cost, *err*: total error %, (err_{GI}): false rejection %, (err_{IG}): false acceptance %).

Method	AR				FERET			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
SRC [3]	246	6.35	10.12	2.84	168	7.14	12.18	1.65
D-KSVD [10]	317	8.86	11.23	3.67	215	9.78	16.73	2.02
LC-KSVD [16]	193	5.53	9.52	2.13	130	6.65	11.19	1.13
FDDL [18]	158	5.21	9.83	1.62	114	5.91	11.04	0.94
LDL [17]	148	4.98	9.27	1.51	108	5.46	**10.37**	0.90
JDDRDL [14]	214	5.78	10.77	2.39	135	6.53	11.46	1.21
DSRC [13]	252	6.16	10.52	2.94	175	7.05	14.07	1.71
CS-JFDL	**102**	**4.40**	**8.98**	**0.88**	**69**	**4.82**	10.82	**0.32**

Table 3
Comparison of cost-sensitive methods on AR and FERET datasets (*cost*: total cost, *err*: total error %, (err_{GI}): false rejection %, (err_{IG}): false acceptance %).

Method	AR				FERET			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
CS-LDA [37]	195	6.35	11.43	2.29	112	7.23	17.32	0.58
CS-KLR [38]	176	6.22	10.86	2.00	100	7.06	16.71	0.43
SCS-C [39]	245	8.06	12.33	2.62	144	9.72	15.76	0.85
CS-LDA + CS-NN	186	6.17	11.22	2.10	94	7.04	18.15	0.46
CSDL [40]	145	5.93	12.80	1.27	97	6.79	16.38	0.42
CS-JFDL	**102**	**4.40**	**8.98**	**0.88**	**69**	**4.82**	**10.82**	**0.32**

ing for SRC (DSRC) [13] and Joint Discriminative Dimensionality Reduction and Dictionary Learning (JDDRDL) [14]. CS-JFDL, DSRC and JDDRDL use the original images for training set (i.e., learn dictionaries using the original raw pixels as the initial image representation) and set the feature dimension after projection as 300. All the other methods use the 300-dimensional Eigenface feature.

We measure the total cost (cost), total error rate (err), error rate of false rejection (err_{GI}) and error rate of false acceptance (err_{IG}). Table 2 reports the average results of different algorithms on AR and FERET datasets. From Table 2, we can observe that CS-JFDL has much smaller total cost than cost-blind dictionary learning methods. It is evident that CS-JFDL achieves this by exploiting cost information of samples during the feature and dictionary learning stage, our CS-JFDL will lead to lower total cost. In addition, CS-JFDL also achieves the best recognition performance and outperforms other two joint feature and dictionary learning methods (DSRC and JDDRDL). The main reason may be that we introduce an intrinsic graph and a penalty graph during the feature learning process such that the learned feature space contains more discriminative information which is useful to classification.

Comparison with SoA Cost-Sensitive Methods: We compare our method with some cost-sensitive feature learning method: Cost-Sensitive Linear Discriminant Analysis (CS-LDA) [37] and Cost-Sensitive classification methods: cost-sensitive kernel logistic regression (CS-KLR) [38] Sparse Cost-Sensitive Classifier (SCS-C) [39] and Cost-Sensitive Dictionary Learning method (CSDL) [40]. Due to extracting cost information in both the feature learning and classification phases can further reduce the total cost [37]. Thus, for a fair comparison, we combine CS-LDA with cost-sensitive nearest neighbor (CS-NN) [38] in this experiment (CSLDA + CS-NN). CSLDA is used for feature learning, and CS-NN is applied for classification. For CSDL, we adopt the discriminative reconstruction error for dictionary learning just as our approach provided in Section 3.3. We perform PCA to learn a linear subspace without cost information, and then performed dictionary learning on these PCA features for face recognition. For CS-LDA, CS-KLR, SCS-C and CS-LDA + CS-NN, we follow the setting of the corresponding papers.

Recognition results of different cost-sensitive methods are listed in Table 3. It can be seen that most of cost-sensitive methods achieve much lower cost and our method CS-JFDL achieves the smallest total cost. Compare with CSDL, CS-JFDL obtains much lower total cost, this indicates that in the reduced cost-sensitive feature space, learn the cost-sensitive dictionary can further reduce the total cost. CS-LDA +CS-NN achieves lower cost than CS-LDA, this clearly indicates that CS-LDA can present a low lost feature distribution for a cost-sensitive classifier to decrease misclassification cost. In spite of that our method still achieves better performance in terms of the total cost.

Joint Learning vs Separate Learning of Feature and Dictionary: The cost-sensitive feature learning and cost-sensitive dictionary can also be learned in an independent manner, i.e., we first learn the cost-sensitive projection matrix from the training data, and then we learn the cost-sensitive dictionary in the reduced cost-sensitive spaces. Denote the independent feature and dictionary learning method as CSFL + CSDL. To show the effect of CS-JFDL, we compare our CS-JFDL with CSFL + CSDL. Table 4 lists the average recognition results. The joint learning manner CS-JFDL achieves lower total cost than independent manner, which indicates that simultaneously learning the features and dictionary is optimal, since some useful information for dictionary learning may be lost in the feature learning stage in independent manner. We can also see that CSFL + CSDL is better than CSDL, this indicates that cost-sensitive features indeed improves the performance of dictionary learning. When the cost-sensitive features and dictionary we learn are combined, the total cost can be further decreased.

Influence of Difference of Learning terms: We investigate the contributions of different terms in our CS-JFDL model. We define the following two alternative baselines to study the importance of different terms in our CS-JFDL models:

(1) CS-JFDL-1: learning the model without cost-sensitive discriminative projection term $G(\boldsymbol{P})$.
(2) CS-JFDL-2: learning the model without cost-sensitive term $H(\boldsymbol{X})$.

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

Table 4
Comparison of different feature and dictionary learning strategies.

Method	AR				FERET			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
CSDL [40]	145	5.93	12.80	1.27	97	6.79	16.38	0.42
CSFL + CSDL	127	5.45	**10.36**	1.12	84	6.13	12.74	0.39
CS-JFDL	**102**	**4.40**	8.98	**0.88**	**69**	**4.82**	**10.82**	**0.32**

Table 5
Comparison of CS-JFDL with different terms.

Method	AR				FERET			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
CS-JFDL-1	121	4.83	9.76	1.10	77	5.45	11.41	0.37
CS-JFDL-2	129	5.21	10.45	1.17	85	5.89	13.22	0.40
CS-JFDL	**102**	**4.40**	**8.98**	**0.88**	**69**	**4.82**	**10.82**	**0.32**

Table 5 shows performance comparisons of CS-JFDL when λ_1 or λ_2 are set as 0 to learn the model, respectively. We can see that cost-sensitive term $H(\boldsymbol{X})$ is more important than discriminative projection term $G(\boldsymbol{P})$ in final performance. Moreover the highest recognition rate can be obtained when both $H(\boldsymbol{X})$ and $G(\boldsymbol{P})$ are used together to learn the model.

In order to further evaluate CS-JFDL, we apply it in more challenging face recognition tasks LFWa [46] and FRGC [47]. For LFWa dataset, we follow the same protocol used in [17], histogram of uniform-LBP is extracted by partitioning a face image into 10×8 patches and the dimension is reduced to 1000. For FRCG dataset, PCA is employed to reduce the dimension as 200. Table 6 summarizes the experimental results. Similar to the results on AR and FERET datasets, it is evident that CS-JFDL achieves this by extracting features that can prevent high-cost errors (err_{IG}). CS-JFDL err_{IG} is lower than other methods err_{IG} on LFWa and FRGC datasets. The results indicate that CS-JFDL can preferentially decrease the high-cost errors, which leads to lower total cost. Especially, CS-JFDL also obtains the competitive classification accuracy.

6.1.3. Parameter analysis

In this section, we first evaluate the performance of CS-JFDL versus different number of gallery classes (M) on AR and FERET datasets. For AR dataset, M varies from 10 to 60 at intervals of 10, and on FERET dataset, M varies from 10 to 70 at intervals of 10. For the sake of simplicity, we only compare our approach with some advanced learning methods. Fig. 5 shows the recognition results of different methods on these two datasets. We can clearly see that our proposed CS-JFDL consistently outperforms other dictionary learning methods under different number of M and obtains the smallest total cost.

We then verify the effect of CS-JFDL under different feature dimensions. Fig. 6(a) shows the total cost of CS-JFDL under different dimensions on the AR dataset. It can be observed that CS-JFDL can achieve the smallest total cost when the feature dimension reaches 180.

Then we investigate the accuracy of CS-JFDL with different dictionary size. Fig. 6(b) shows the total cost of our method with different dictionary size on the AR dataset. We see that with the increase of the number of dictionary size, the performance of CS-JFDL also increases.

For discussing the influence of parameter σ, we evaluate CS-JFDL with different σ on AR and FERET datasets. We varies σ from 1 to 12. The experiments are repeated 10 times. Fig. 7 shows the total cost and classification error of CS-JFDL on different datasets. From Fig. 7 we can see that the total cost varies only within a small range, and the recognition error varies with a large change relatively with the increase of σ. Thus, in order to obtain a smaller cost, in all experiments, we set $\sigma = 2$.

6.1.4. Discussion

From the above experimental results, we see that our CS-JFDL method achieves better performance and consistently outperforms the CSDL [40]. The main reason can be summarized as follows:

- CSDL considers feature learning and dictionary are two independent problems. Thus, CSDL fails to capture the relationship between the features and the dictionary. Furthermore, this will result in loss of key discriminative information for classification during the learning process and the learned dictionary may not be optimal. While our method adopts a novel joint learning technique to build discriminative features and structured dictionaries simultaneously such that the learned features and dictionary are complementary to each other.
- Our method puts the cost information into use during the feature learning phase, while CSDL just applies a cost-blind dimensionality reduction, i.e. principal component analysis (PCA), as the prior before dictionary learning.

6.1.5. Learning the cost matrix

In this paper, we assume that the cost matrix is given by the user and can reflect user's security consideration. However, in many cases, how to select appropriate measure and provide clear cost ratios is difficult for the users. Thus, refining the cost matrix given by users or learning a cost matrix via the interaction with users is desired for a cost-sensitive system. However,

Table 6
Comparisons on LFWa and FRGC datasets.

Method	LFWa				FRGC			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
SRC [3]	1130	27.32	29.53	12.56	829	13.16	14.54	7.06
LC-KSVD [16]	1098	25.56	33.73	12.04	624	9.95	12.18	5.23
FDDL [18]	1003	22.53	29.50	11.16	662	10.11	11.87	5.67
LDL [17]	932	21.8	**26.74**	10.42	634	9.07	11.21	5.47
JDDRDL [14]	1075	25.16	29.33	12.08	733	10.33	12.54	6.43
CSDL [40]	802	22.73	31.54	8.02	421	10.27	11.56	2.85
CS-JFDL	**699**	**20.48**	28.89	**6.84**	**284**	**8.31**	**9.88**	**1.62**

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

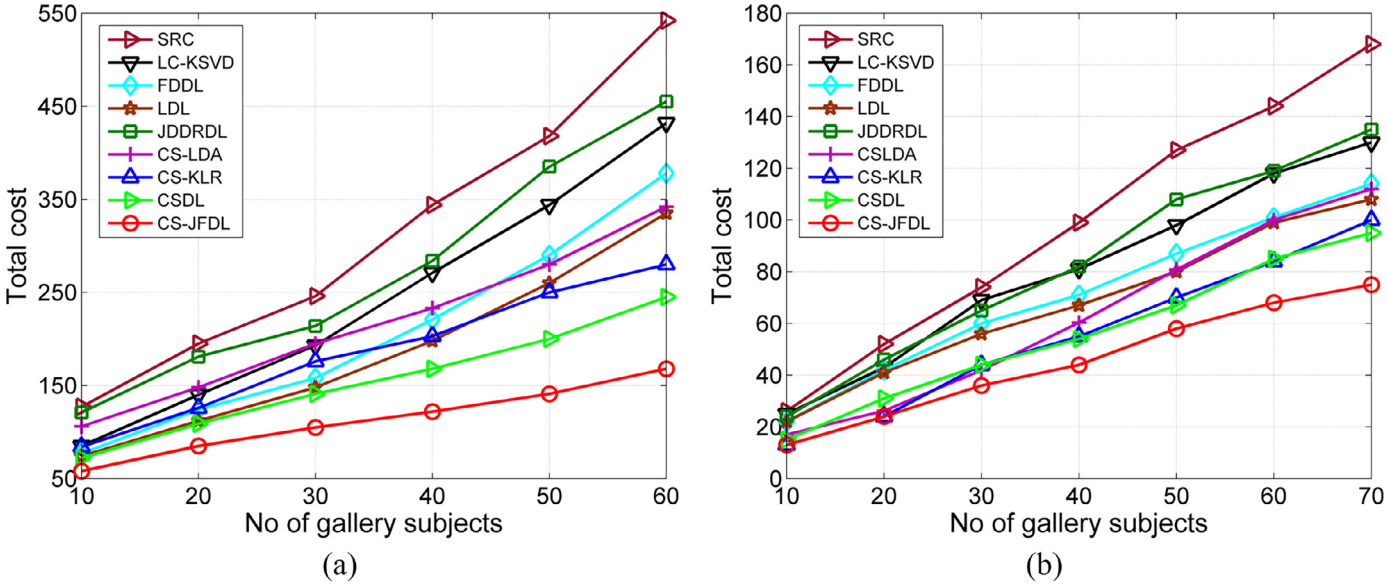

Fig. 5. Comparison of different methods under different number of gallery subjects on (a)AR dataset, (b) FERET dataset.

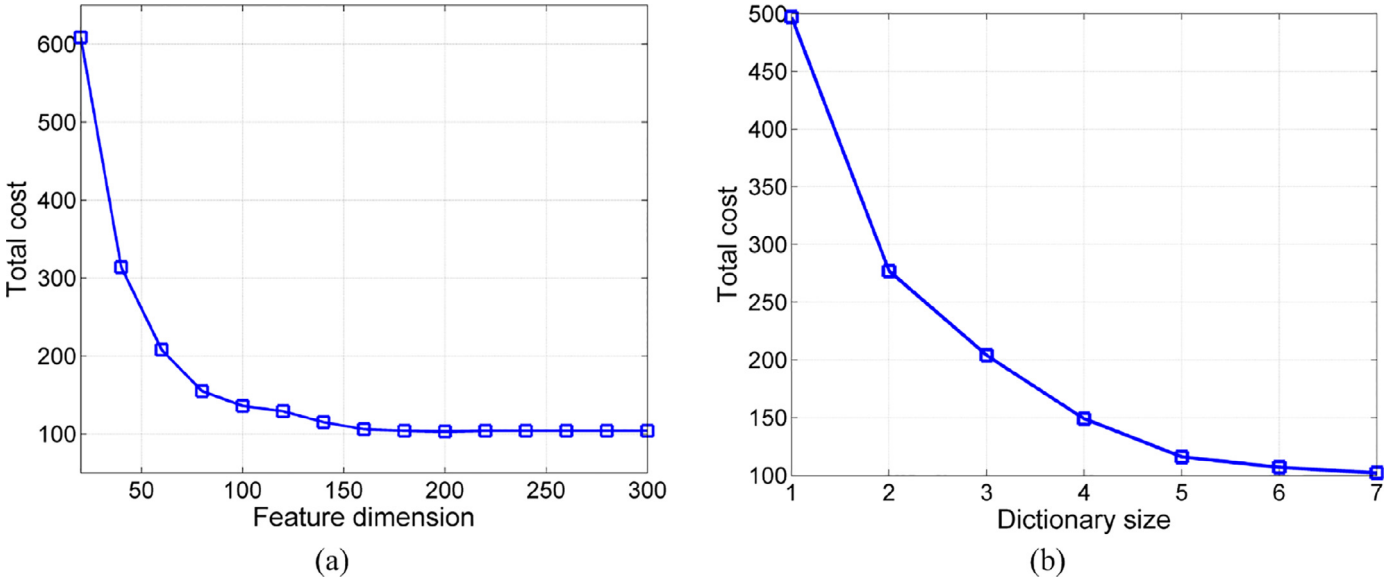

Fig. 6. Total cost of our CS-JFDL versus different feature dimensions and different number of atoms on the AR dataset. (a) feature dimension, (b)dictionary size.

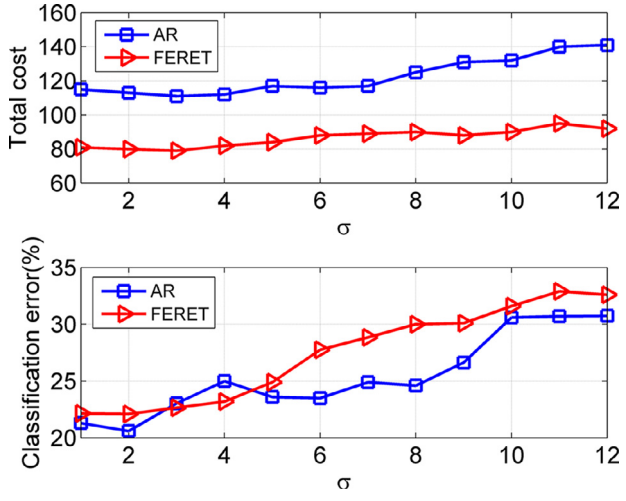

Fig. 7. The top row shows total cost of CS-JFDL versus different σ. The bottom row shows classification error versus different σ.

to the best our knowledge, there is not a good method to learn the cost matrix, and it is still an open and challenging problem. Zhang et al. [32] exploited an efficient way to learn the cost matrix, where only one classifier is needed to be trained for obtaining results for all parameter settings. In this paper, we only focus on how to develop a appropriate algorithm to reduce the final classification losses. In addition, Designing cost matrix is time-consuming. Therefore, learning cost matrix is not the key point in our paper. If readers are interested in it, please refer to [32]. In order to facilitate research in the area, a Matlab implementation of our method will be made available.

6.2. Image set based face recognition

Applying the proposed classification scheme in Section 5, our method can be extended to handle image set classification [53–58]. Given a testing video $\boldsymbol{Y}^{te} = [\boldsymbol{y}_1^{te}, \boldsymbol{y}_2^{te}, \ldots, \boldsymbol{y}_{N_{te}}^{te}]$, where $\boldsymbol{y}_j^{te}$ is the jth $(1 \le j \le N_{te})$ frame of this video and N_{te} is the number of image frames in this video, we first apply the learned feature projection matrix $\boldsymbol{P}$ to project each frame $\boldsymbol{y}_j^{te}$ to a feature and predict its la-

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

bels by using the smallest reconstruction error corresponding to each sub-dictionary $\boldsymbol{D}_i$ $(1 \leq i \leq c)$, as described in Eq. (15). After getting all the labels of frames, we perform a majority voting to decide the label of the given image set. For testing efficiency, we use the l_2 norm $\|\boldsymbol{x}\|_2$ instead of l_1 norm $\|\boldsymbol{x}\|_1$ and derive the decision:

$$label(\boldsymbol{y}_j^{te}) = \arg\min_i \left\{ \|\boldsymbol{P}\boldsymbol{y}_j^{te} - \boldsymbol{D}_i\delta_i(\boldsymbol{D}^{\dagger}\boldsymbol{y}_j^{te})\|_2^2 \right\} \tag{14}$$

where $\boldsymbol{D}^{\dagger} = (\boldsymbol{D}^T\boldsymbol{D} + \alpha_2\boldsymbol{I})^{-1}\boldsymbol{D}^T$ is the pseudoinverse of $\boldsymbol{D}$

Then, we adopt the majority voting strategy to classify the whole testing video (image set):

$$label(\boldsymbol{Y}^{te}) = \arg\min_i H_i \tag{15}$$

where H_i is the total number of votes from the ith class.

In this section, we use Honda/UCSD [48], CMU Mobo [49], and Youtube Celebrities (YTC) [50] datasets to evaluate the performance of CS-JFDL. The Honda/UCSD dataset includes 59 face videos involving 20 individuals with large pose and expression variations. The average lengths of these videos are approximately 400 frames. The CMU MoBo dataset includes 96 videos from 24 individuals. For each subject, there four videos corresponding to different walking patterns. For each video, there are around 300 frames. The YTC dataset includes 1910 videos of 47 celebrities from YouTube. Most videos contain noisy and low-quality image frames. The number of frames in a video varies from 8 to 400.

For face videos in the Honda, Mobo and YTC datasets, all image frames are detected using the face detector method proposed in [52] and then resize them to 30×30 intensity image. Thus each video is represented as an image set. For each image frame in all these three datasets, we only perform histogram equalization but no further pre-processing and the image features are raw pixel values.

On Honda, MoBo, and YTC datasets, we randomly select training and testing sets 10 times, then compute and compare the average recognition performance. For the Honda dataset, one video per subject is randomly selected for training, while the remaining as testing. Specifically, In the training set, we use 10 video sequences as the gallery subjects and the rest 10 video sequences as the impostor subjects, and select N_G frames from each image set for training. The rest videos corresponding to each chosen subject are used as the testing set. For the MoBo dataset, we randomly select one face video per person for training and the use the rest videos for testing. In the training set, half of them are selected as the gallery subjects and the remaining half as the impostor subjects and each image set select N_G frames for training. For the YTC dataset, we equally divide the whole dataset into five folds, and each fold contains 9 videos for each person. In each fold, we randomly select 20 gallery subjects and 20 impostor subjects. In the training set, 3 face videos for each person are used for training, and each image set select N_G frames. In the testing set, the remain 6 face videos for each person are used for testing, and each image set also select N_G frames. $\boldsymbol{C}_{GI}$, $\boldsymbol{C}_{IG}$, and $\boldsymbol{C}_{GG}$ are specified as the same in Table 1.

In our experiments, the feature dimension of $\boldsymbol{P}$ is specified as 200. We fix $\lambda_1 = 1, \lambda_2 = 2, \lambda_3 = 0.05, \alpha_2 = 0.001, k_1 = 5$ and $k_2 = 30$ respectively. The number of atoms per subject are set as 20, 25 and 35 on Honda/UCSD, CMU MoBo and YTC, respectively.

Comparison with SoA Image set Based approaches: we compare CS-JFDL with several image set based recognition approaches, including Manifold-to-Manifold Distance (MMD) [53], Manifold Discriminant Analysis (MDA) [54], Convex Hull based Image Set Distance (CHISD) [55], Sparse Approximated Nearest Point (SANP) [56], Local Multi-Kernel Metric Learning (LMKML) [57], Projection Metric Learning (PML) [58], and Simultaneous Feature and Dictionary Learning (SFLD) [20]. The settings of these approaches are basically the same as [20]. For Honda/UCSD and CMU MoBo datasets, we randomly select 50 and 200 frames for training, respectively. Table 7 and Table 8 list the experimental results of different methods on Honda, and MoBo datasets. We observe that our CS-JFDL achieves the lowest total cost and consistently outperforms other methods on all the experiments. The main reason is that our method can exploit more discriminative information in the learned feature subspace and the learned structured dictionary can extract more person-specific information.

We also evaluate the performance of CS-JFDL on YTC dataset when videos contain different number of image frames. We randomly select N_G frames from each image set for training and use another N_G frames for recognition. Note that if there is an image set that do not have enough image frames, we use all of the frames in the image set instead. Fig. 8 shows the experimental results of different methods with varying image frames. One can see that CS-JFDL achieves smaller cost than the other approaches. This indicates that our method is effective for image set recognition in terms of the total cost.

Table 7
Comparison on Honda/UCSD dataset with different frames.

Method	50Frames				200Frames			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
MMD [53]	401	19.22	20.5	7.89	23	2.56	2.11	1.5
MDA [54]	227	9.23	10.53	4.50	45	1.28	1.0	1.05
CHISD [55]	224	8.97	9.47	4.50	116	3.85	3.0	2.63
SANP [56]	174	6.92	7.00	3.68	94	3.33	2.5	2.11
LMKML [57]	150	5.89	6.50	3.16	23	0.77	0.53	0.5
PML [58]	102	4.36	4.74	2.00	**0**	**0**	**0**	**0**
SFDL [20]	81	**4.10**	**4.01**	1.58	**0**	**0**	**0**	**0**
CS-JFDL	**44**	**4.10**	4.50	**0.53**	**0**	**0**	**0**	**0**

Table 8
Comparison on CMU MoBo dataset with different frames.

Method	50Frames				200Frames			
	cost	*err*	err_{GI}	err_{IG}	*cost*	*err*	err_{GI}	err_{IG}
MMD [53]	256	5.83	6.67	2.78	207	4.86	5.56	2.22
MDA [54]	315	7.64	8.89	3.33	153	3.47	3.89	1.67
CHISD [55]	198	4.17	4.44	2.22	126	2.78	3.06	1.39
SANP [56]	176	3.89	4.17	1.94	106	2.64	2.78	1.13
LMKML [57]	156	3.75	4.17	1.67	123	2.5	2.78	1.39
PML [58]	172	3.47	3.89	1.94	84	2.22	3.06	0.83
SFDL [20]	131	**3.19**	**3.61**	1.39	71	1.94	2.32	0.71
CS-JFDL	**60**	3.33	4.72	**0.28**	**29**	**1.81**	**2.5**	**0.11**

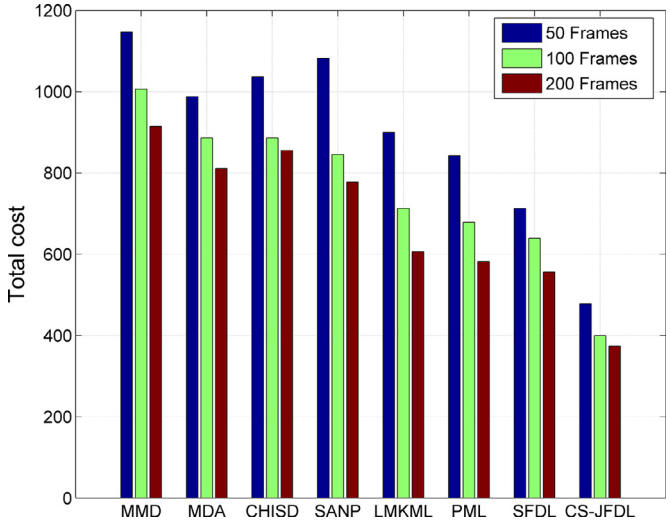

Fig. 8. Total cost with different number of image frames on YTC dataset.

7. Conclusion

We presented a novel cost-sensitive joint feature and dictionary learning method (CS-JFDL). By jointly learning the discriminative projection matrix and the structured dictionary, our method extracts more discriminative information for classification. In addition, we introduce the cost information of samples into the feature and dictionary learning stage and enforce the cost-sensitive requirement throughout the entire learning process. Unlike most existing dictionary learning algorithms which do not consider the cost information, our method achieved lower total cost than traditional dictionary learning methods. Extensive experimental evaluations show that CS-JFDL achieves superior performance on both image based face recognition and image set based face recognition tasks.

Declaration of Competing Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

CRediT authorship contribution statement

Guoing Zhang: Conceptualization, Methodology, Software, Resources, Data curation, Writing - original draft, Visualization, Supervision, Project administration, Funding acquisition. **Fatih Porikli:** Methodology, Formal analysis, Writing - review & editing. **Huaijiang Sun:** Methodology, Formal analysis, Writing - review & editing, Funding acquisition. **Quansen Sun:** Methodology, Supervision, Funding acquisition. **Guiyu Xia:** Validation. **Yuhui Zheng:** Resources, Supervision.

Acknowledgment

This work was supported by the National Nature Science Foundation of China under Grants 61772272, 61,673,220 and 61806099, in part by the Natural Science Foundation of Jiangsu Province, China under Grant BK20180790, in part by the Natural Science Research of Jiangsu Higher Education Institutions of China under Grant 18KJB520033F. Porikli was supported in part under the Australian Research Council's Discovery Projects funding scheme (project DP150104645).

Appendix A. Optimization

In this appendix, the optimization procedure of Eq. (11) is provided.

Step 1: Learn the cost-sensitive discriminative projection.

In order to learn projection $\boldsymbol{P}$, we fix $\boldsymbol{D}$ and $\boldsymbol{X}$. Let $\boldsymbol{V} = \boldsymbol{DX}$, $\boldsymbol{V}_i = \boldsymbol{DX}_i^i$, then Eq. (11) can be expressed as

$$\min_{\boldsymbol{P}} J = \|\boldsymbol{PY} - \boldsymbol{V}\|_F^2 + \sum_{i=1}^{c} \|\boldsymbol{PY}_i - \boldsymbol{V}_i\|_F^2 - \lambda_1 tr\left(\boldsymbol{PYL}^2\boldsymbol{Y}^T\boldsymbol{P}^T\right), \quad s.t. \boldsymbol{PP}^T = \boldsymbol{I}. \tag{A.1}$$

We can see that Eq. (A.1) is non-convex, and we can have a local minimum of it as follows. Since $\boldsymbol{PP}^T = \boldsymbol{I}$, we get

$$\|\boldsymbol{PY} - \boldsymbol{V}\|_F^2 = tr\left(\boldsymbol{P}\varphi(\boldsymbol{P})\boldsymbol{P}^T\right) \tag{A.2}$$

and

$$\sum_{i=1}^{c} \|\boldsymbol{PY}_i - \boldsymbol{V}_i\|_F^2 = tr\left(\boldsymbol{P}\varphi(\boldsymbol{P}_i)\boldsymbol{P}^T\right) \tag{A.3}$$

where $\varphi(\boldsymbol{P}) = (\boldsymbol{Y} - \boldsymbol{P}^T\boldsymbol{V})(\boldsymbol{Y} - \boldsymbol{P}^T\boldsymbol{V})^T$, $\varphi(\boldsymbol{P}_i) = \sum_{i=1}^{c}(\boldsymbol{Y} - \boldsymbol{P}^T\boldsymbol{V}_i)(\boldsymbol{Y} - \boldsymbol{P}^T\boldsymbol{V}_i)^T$. Then Eq. (A.1) can be rewritten as

$$\min_{\boldsymbol{P}} J = tr\left(\boldsymbol{P}\varphi(\boldsymbol{P})\boldsymbol{P}^T\right) + tr\left(\boldsymbol{P}\varphi(\boldsymbol{P}_i)\boldsymbol{P}^T\right) - \lambda_1 tr\left(\boldsymbol{PYL}^2\boldsymbol{Y}^T\boldsymbol{P}^T\right), \quad = tr\left(\boldsymbol{P}(\varphi(\boldsymbol{P}) + \varphi(\boldsymbol{P}_i) - \lambda_1\boldsymbol{YL}^2\boldsymbol{Y}^T)\boldsymbol{P}^T\right), \quad s.t. \boldsymbol{PP}^T = \boldsymbol{I}. \tag{A.4}$$

In the current iteration t, to obtain the above minimization, we exploit $\varphi(\boldsymbol{P}_{(t-1)})$ and $\varphi(\boldsymbol{P}_{i,(t-1)})$ to approximate $\varphi(\boldsymbol{P})$ and $\varphi(\boldsymbol{P}_i)$ in Eq. (A.4), where $\boldsymbol{P}_{(t-1)}$ is the projection obtained in iteration $t-1$. We use Eigen Value Decomposition (EVD) technique to get

$$[\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{U}] = EVD\left(\varphi(\boldsymbol{P}_{(t-1)}) + \varphi(\boldsymbol{P}_{i,(t-1)}) - \lambda_1\boldsymbol{YL}^2\boldsymbol{Y}^T\right) \tag{A.5}$$

where $\boldsymbol{\Sigma}$ is diagonal matrix formed by the eigenvalues of $(\varphi(\boldsymbol{P}_{(t-1)}) + \varphi(\boldsymbol{P}_{i,(t-1)}) - \lambda_1\boldsymbol{YL}^2\boldsymbol{Y}^T)$. Then set $\boldsymbol{P}$ as the matrix of eigenvectors in $\boldsymbol{U}$ corresponding to the first d eigenvalues, i.e., let $\boldsymbol{P}_{(t-1)} = \boldsymbol{U}(1:d,:)$. Nevertheless, by this means the update of $\boldsymbol{P}$ probably too big, and the optimization of the whole objective function in Eq (11) may be unstable. Thus, we update $\boldsymbol{P}$ gradually in each iteration and denote

$$\boldsymbol{P}_{(t)} = \boldsymbol{P}_{(t-1)} + o(\boldsymbol{U}(1:d,:) - \boldsymbol{P}_{(t-1)} \tag{A.6}$$

where o is a small positive constant to control the change of $\boldsymbol{P}$ in iterations.

Step 2: Learn the sparse coding matrix.

To learn $\boldsymbol{X}$, we fix $\boldsymbol{P}$ and $\boldsymbol{D}$, then Eq. (11) can be rewritten as

$$\min_{\boldsymbol{X}} J = \sum_{i=1}^{c}(\|\boldsymbol{PY}_i - \boldsymbol{DX}_i\|_F^2 + \|\boldsymbol{PY}_i - \boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2 + \sum_{j=1,j\neq i}^{c}\|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2) + \lambda_2\|\boldsymbol{Q}\odot\boldsymbol{X}\|_F^2 + \lambda_3\|\boldsymbol{X}\|_1 \tag{A.7}$$

We compute $\boldsymbol{X}_i$ sequentially by fixing other coefficient matrices $\boldsymbol{X}_j$ $(j \neq i\, 1 \leq i \leq c)$. Thus Eq. (A.7) can be simplified as

$$\min_{\boldsymbol{X}_i} J = \|\boldsymbol{PY}_i - \boldsymbol{DX}_i\|_F^2 + \|\boldsymbol{PY}_i - \boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2 + \sum_{j=1,j\neq i}^{c}\|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2 + \lambda_2\|\boldsymbol{Q}_i\odot\boldsymbol{X}_i\|_F^2 + \lambda_3\|\boldsymbol{X}_i\|_1 \tag{A.8}$$

Following [15,20,42], we optimize each $\boldsymbol{x}_{is}$ in $\boldsymbol{X}_i$. We define $\boldsymbol{x}_{is}$ as the coding coefficient of the s-th sample in the ith class. For obtaining $\boldsymbol{x}_{is}$, we fix other coding coefficients $\boldsymbol{x}_{it}$ $(t \neq s)$ for other samples and rewrite Eq. (A.8) as

$$\min_{\boldsymbol{x}_{is}} J = R(\boldsymbol{D}, \boldsymbol{P}, \boldsymbol{x}_{is}) + \lambda_2\boldsymbol{x}_{is}^T diag(\boldsymbol{q}_{is})^2\boldsymbol{x}_{is} + \lambda_3\sum_{z=1}^{c}|\boldsymbol{x}_{is}^{(z)}| \tag{A.9}$$

where

$$R(\boldsymbol{D}, \boldsymbol{P}, \boldsymbol{x}_{is}) = \|\boldsymbol{Py}_{is} - \boldsymbol{Dx}_{is}\|_2^2 + \|\boldsymbol{Py}_{is} - \boldsymbol{D}_i\boldsymbol{x}_{is}^i\|_2^2 + \sum_{j=1,j\neq i}^{c}\|\boldsymbol{D}_j\boldsymbol{x}_{is}^j\|_2^2 \tag{A.10}$$

where $diag(\boldsymbol{q}_{is})$ is a diagonal matrix with (z, z)-th element as the zth entry of $\boldsymbol{q}_{is}$ and $\boldsymbol{x}_{is}^{(z)}$ is the (z, z)-th component of $\boldsymbol{x}_{is}$. Eq. (A.9) can be solved using feature sign search algorithm [43] after certain formulation based on [15,20].

Step 3: Learn the dictionary.

By fixing $\boldsymbol{P}$ and $\boldsymbol{X}$, we can learn $\boldsymbol{D}$, Eq. (11) can be rewritten as

$$\min_{\boldsymbol{D}} J = \sum_{i=1}^{c}(\|\boldsymbol{PY}_i - \boldsymbol{DX}_i\|_F^2 + \|\boldsymbol{PY}_i - \boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2 + \sum_{j=1,j\neq i}^{c}\|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2) \tag{A.11}$$

We update $\boldsymbol{D}$ class by class sequentially. When updating $\boldsymbol{D}_i$, the sub-dictionaries $\boldsymbol{D}_j$, $j \neq i$ associated to other classes will be fixed.

Thus Eq. (A.11) can be further rewritten as

$$\min_{\boldsymbol{D}_i} J = \|\boldsymbol{PY}_i - \boldsymbol{DX}_i\|_F^2 + \|\boldsymbol{PY}_i - \boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2 + \sum_{j=1, j\neq i}^{c} \|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2. \quad \text{(A.12)}$$

Which essentially a quadratic programming problem and can be directly solved by the algorithm presented in [41] (update $\boldsymbol{D}_i$ atom by atom). Notice that each atom in the dictionary should have unit l_2 norm.

References

[1] W. Zhao, R. Chellppa, P.J. Phillips, A. Rosenfeld, Face recognition: a literature survey, ACM Comput. Surv. 35 (1) (2003) 399–458.
[2] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenface vs. fisherfaces: recognition using class specific linear projection, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (2003) 711–720.
[3] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[4] J. Yang, D. Chu, L. Zhang, Y. Xu, J. Yang, Sparse representation classifier steered discriminative projection with applications to face recognition, IEEE Trans. Neural Netw. Learn. Syst. 24 (7) (2013) 1023–1035.
[5] Y. Gao, J. Ma, A.L. Yuille, Semi-supervised sparse representation based classification for face recognition with insufficient labeled samples, IEEE Trans. Image Process. 26 (5) (2017) 2545–2560.
[6] J. Lu, V.E. Liong, X. Zhou, J. Zhou, Learning compact binary face descriptor for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (10) (2015) 2041–2056.
[7] Y. Duan, J. Lu, J. Feng, J. Zhou, Context-aware local binary feature learning for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2017) 1864–1870.
[8] J. Lu, V.E. Liong, J. Zhou, Simultaneous local binary feature learning and encoding for face recognition, in: Proceedings of the IEEE 15th International Conference on Computer Vision (ICCV), 2015, pp. 3721–3729.
[9] Q. Mao, Q. Rao, Y. Yu, M. Dong, Hierarchical bayesian theme models for multipose facial expression recognition, IEEE Trans. Multim. 19 (4) (2017) 861–873.
[10] Q. Zhang, B. Li, Discriminative K-SVD for dictionary learning in face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2691–2698.
[11] G. Zhang, H. Sun, G. Xia, Q. Sun, Kernel collaborative representation based dictionary learning and discriminative projection, Neurocomputing 207 (2016) 300–309.
[12] G. Zhang, H. Sun, G. Xia, L. Feng, Q. Sun, Kernel dictionary learning based discriminant analysis, J. Vis. Commun. Image Represent. 40 (2016) 470–484.
[13] H. Zhang, Y. Zhang, T.S. Huang, Simultaneous discriminative projection and dictionary learning for sparse representation based classification, Pattern Recognit. 46 (1) (2013) 346–354.
[14] Z. Feng, M. Yang, L. Zhang, Y. Liu, D. Zhang, Joint discriminative dimensionality reduction and dictionary learning for face recognition, Pattern Recognit. 46 (8) (2013) 2134–2143.
[15] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Trans. Image Process. 20 (5) (2011) 1327–1336.
[16] Z. Jiang, L. Davis, Label consistent k-SVD: learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2651–2664.
[17] M. Yang, D. Dai, L. Shen, L. Gool, Latent dictionary learning for sparse representation based classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 4138–4145.
[18] M. Yang, L. Zhang, X. Feng, D. Zhang, Fisher discrimination dictionary learning for sparse representation, in: Proceedings of the IEEE 15th International Conference on Computer Vision (ICCV), 2011, pp. 543–550.
[19] M. Yang, W. Liu, W. Luo, L. Shen, Analysis-synthesis dictionary learning for universality-particularity representation based classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2016, pp. 2251–2257.
[20] J. Lu, G. Wang, J. Zhou, Simultaneous feature and dictionary learning for image set based face recognition, IEEE Trans. Image Process. 26 (8) (2017) 4042–4054.
[21] G. Zhang, H. Sun, F. Porikli, Y. Liu, Q. Sun, Optimal couple projections for domain adaptive sparse representation-based classification, IEEE Trans. Image Process. 26 (12) (2017) 5922–5935.
[22] W. Ouyang, X. Wang, A discriminative deep model for pedestrian detection with occlusion handling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3258–3265.
[23] Q. Le, A. Karpenko, J. Ngiam, A. Ng, ICA with reconstruction cost for efficient overcomplete feature learning, in: Proceedings of the Advances in Neural Information Processing Systems (NIPS), 2011, pp. 1017–1025.
[24] Q. Le, W. Zou, S. Yeung, A. Ng, Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3361–3368.
[25] J. Fan, W. Xu, Y. Wu, Y. Gong, Human tracking using convolutional neural networks, IEEE Trans. Neural Netw. 21 (10) (2010) 1610–1623.
[26] T. Xiao, S. Li, B. Wang, L. Lin, X. Wang, Joint detection and identification feature learning for person search, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3415–3424.
[27] Z. Cao, Q. Yin, X. Tang, J. Sun, Face recognition with learning based descriptor, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2707–2714.
[28] Z. Lei, M. Pietikäinen, S.Z. Li, Learning discriminant face descriptor, IEEE Trans. Pattern Anal. Mach. Intell. 36 (3) (2014) 289–302.
[29] J. Lu, V.E. Liong, J. Zhou, Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 40 (8) (2018) 1979–1993.
[30] W. Huang, F. Sun, L. Cao, D. Zhao, H. Liu, M. Harandi, Sparse coding and dictionary learning with linear dynamical systems, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3938–3947.
[31] G. Zhang, H. Sun, G. Xia, Q. Sun, Multiple kernel sparse representation based orthogonal discriminative projection and its cost-sensitive extension, IEEE Trans. Image Process. 25 (9) (2016) 4271–4285.
[32] S.H. Khan, M. Bennamoun, F.A. Sohel, R. Togneri, Cost sensitive learning of deep feature representation from imbalanced data, IEEE Trans. Neural Netw. Learn. Syst. 29 (8) (2018) 3573–3587.
[33] J. Lu, V.E. Liong, J. Zhou, Cost-sensitive local binary feature learning for facial age estimation, IEEE Trans. Image Process. 24 (12) (2015) 5356–5368.
[34] S. Feng, C. Lang, J. Feng, T. Wang, J. Luo, Human facial age estimation by cost-sensitive label ranking and trace norm regularization, IEEE Trans. Multim. 19 (1) (2017) 136–148.
[35] J. Lu, X. Zhou, Y. Tan, Y. Shang, J. Zhou, Cost-sensitive semi-supervised discriminant analysis for face recognition, IEEE Trans. Inf. Forensics Secur. 7 (3) (2012) 944–953.
[36] J. Lu, Y. Tan, Cost-sensitive subspace learning for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2661–2666.
[37] J. Lu, Y. Tan, Cost-sensitive subspace analysis and extensions and face recognition, IEEE Trans. Inf. Forensics Secur. 8 (3) (2013) 510–519.
[38] Y. Zhang, Z. Zhou, Cost-sensitive face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (10) (2010) 1758–1769.
[39] J. Man, X. Jing, D. Zhang, C. Lan, Sparse cost-sensitive classifier with application to face recognition, in: Proceedings of the IEEE Conference on Image Process (ICIP), 2011, pp. 1773–1776.
[40] G. Zhang, H. Sun, Z. Ji, Y. Yuan, Q. Sun, Cost-sensitive dictionary learning for face recognition, Pattern Recognit. 60 (2016) 613–629.
[41] M. Yang, L. Zhang, J. Yang, D. Zhang, Metaface learning for sparse representation based face recognition, in: Proceedings of the IEEE Conference on Image Process (ICIP), 2010, pp. 1601–1604.
[42] W. Liu, Z. Yu, Y. Wen, R. Lin, M. Yang, Jointly learning non-negative projection and dictionary with discriminative graph constraints for classification, arXiv preprint arXiv:1511.04601 (2015).
[43] H. Lee, A. Battle, R. Raina, A. Ng, Efficient sparse coding algorithms, in: Proceedings of the Advances in Neural Information Processing Systems (NIPS), 2006, pp. 801–808.
[44] A.M. Martinez, R. Benavente, The AR face database, CVC, Univ. Autónoma Barcelona, Barcelona, Spain, 1998 Tech. rep. #24. Jun.
[45] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss, The FERET evaluation methodology for face recognition algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 1090–1104.
[46] L. Wolf, T. Hassner, Y. Taigman, Effective unconstrained face recognition by combining multiple descriptors and learned background statistics, IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 1978–1990.
[47] P.J. Phillips, P.J. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, W. Worek, Overview of the face recognition grand challenge, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 947–954.
[48] K.-C. Lee, J. Ho, M.H. Kriegman, Video-base face recognition using probabilistic appearance manifolds, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2003, pp. 313–320.
[49] R. Gross, J. Shi, The CMU motion of body (mobo) database, Tech. Rep. 27 (1) (2001) 1–13.
[50] M. Kim, S. Kumar, V. Pavlovic, H. Rowley, Face tracking and recognition with visual constraints in real-world videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–8.
[51] G. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: a database for studying face recognition in unconstrained environments, University of Massachusetts, Amherst, 2007 Technical report. 1 (2) 07–49
[52] P. Viola, M.J. Jones, Robust real-time face detection, Int. J. Comput. Vis. 57 (2) (2004) 137–154.
[53] R. Wang, S. Shan, X. Chen, W. Gao, Manifold-manifold distance with application to face recognition based on image set, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–8.
[54] R. Wang, X. Chen, Manifold discriminant analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 429–436.
[55] H. Cevikalp, B. Triggs, Face recognition based on image sets, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2567–2573.

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101

[56] Y. Hu, A.S. Mian, R. Owens, Sparse approximated nearest points for image set classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 121–128.
[57] J. Lu, G. Wang, P. Moulin, Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning, in: Proceedings of the IEEE 14th International Conference on Computer Visison (ICCV), 2013, pp. 329–336.
[58] Z. Huang, R. Wang, S. Shan, X. Chen, Projection metric learning on Grassmann manifold with application to video based face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 140–149.
[59] J. Liu, N. Sun, X. Li, G. Han, H. Yang, Q. Sun, Rare bird sparse recognition via part-based gist feature fusion and regularized intraclass dictionary learning, Comput, Mater. Contin. 55 (3) (2018) 435–446.
[60] Y. Duan, J. Lu, Z. Wang, J. Feng, J. Zhou, Learning deep binary descriptor with multi-quantization, IEEE Trans. Pattern Anal. Mach. Intell. 41 (8) (2019) 1924–1938.
[61] H. Liu, H. Liu, F. Sun, B. Fang, Kernel regularized nonlinear dictionary learning for sparse coding, IEEE Trans. Syst., Man Cybern.: Syst. 49 (4) (2019) 766–775.

Guoqing Zhang received the B.S. and Master degrees in information engineering from the Yangzhou University, Yangzhou, China, in 2009 and 2012, and the Ph.D. degree in pattern recognition and intelligence system from Nanjing University of Science and Technology, Nanjing, China, in 2017. He is assistant professor with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China. His main research interests include computer vision, pattern recognition and machine learning.

Fatih Porikli is an IEEE Fellow and a Professor in the Research School of Engineering, Australian National University (ANU). He is also acting as the Chief Scientist at Huawei, Santa Clara. He has received his Ph.D. from New York University in 2002. Previously he served Distinguished Research Scientist at Mitsubishi Electric Research Laboratories. His research interests include computer vision, pattern recognition, manifold learning, image enhancement, robust and sparse optimization and online learning with commercial applications in autonomous vehicles, video surveillance, visual inspection, robotics, consumer electronics, satellite imaging and medical systems. Prof. Porikli is the recipient of the R&D 100 Scientist of the Year Award in 2006. He won 5 best paper awards at premier IEEE conferences. Prof. Porikli authored more than 200 publications, invented 71 US patents, and co-edited 2 books. He is serving as the Associate Editor of 5 journals for the past 10 years.

Huaijiang Sun received the B. Eng. and PhD.degrees in the School of 670 Marine Engineering, Northwestern Polytechnical University, Xi'an, China, in 1990 and 1995, respectively. He is currently a Professor in the Department of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include computer vision and pattern recognition, image and video processing and intelligent information processing.

Quansen Sun received the PhD in Pattern Recognition and Intelligence System from Nanjing University of Science and Technology, Nanjing, China, in 2006. He is currently a professor in the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include pattern recognition, image processing, computer vision and data fusion.

Guiyu Xia received his B.S. and Ph. D. degrees in software engineering from Nanjing University of Science and Technology, Nanjing 210094, China, in 2012 and 2017. Currently, he is a FFaculty Member with the school of Automation, Nanjing University of Information Science and Technology, China. His research interest covers pattern recognition, machine learning and human motion capture data reusing.

Yuhui Zheng received the B.Sc. degree in pharmacy engineering and the Ph.D. degree in pattern recognition and intelligent system from Nanjing University of Science and Technology, Nanjing, China, in 2004 and 2009, respectively. He is currently an associate Professor with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China. His research interests are multimedia data analysis and processing, image and video segmentation, and computer vision.

Please cite this article as: G. Zhang, F. Porikli and H. Sun et al., Cost-sensitive joint feature and dictionary learning for face recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.01.101