Neurocomputing 173 (2016) 1554–1564

Contents lists available at ScienceDirect

Neurocomputing

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

Spatial regularization in subspace learning for face recognition: implicit vs. explicit

Yulian Zhu [a,b], Songcan Chen [b,*], Qing Tian [b]

[a] Computer Center, Nanjing University of Aeronautics & Astronautics, Nanjing 211106, China
[b] College of Computer Science and Technology, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China

ARTICLE INFO

Article history:
Received 12 March 2015
Received in revised form
6 September 2015
Accepted 6 September 2015
Communicated by Ke Lu
Available online 25 September 2015

Keywords:
Face recognition
Implicit spatial regularization (ISR)
Explicit spatial regularization (ESR)
Subspace learning
Linear discriminant analysis (LDA)
Locality preserving projection (LPP)

ABSTRACT

In applying traditional statistical method to face recognition, each original face image is often vectorized as a vector. But such a vectorization not only leads to high-dimensionality, thus small sample size (SSS) problem, but also loses the original spatial relationship between image pixels. It has been proved that spatial regularization (SR) is an effective means to compensate the loss of such relationship and at the same time, and mitigate SSS problem by explicitly imposing spatial constraints. However, SR still suffers from two main problems: one is high computational cost due to high dimensionality and the other is the selection of the key regularization factors controlling the spatial regularization and thus learning performance. Accordingly, in this paper, we provide a new idea, coined as implicit spatial regularization (ISR), to avoid losing the spatial relationship between image pixels and deal with SSS problem simultaneously for face recognition. Different from explicit spatial regularization (ESR), which introduces directly spatial regularization term and is based on vector representation, the proposed ISR constrains spatial smoothness within each small image region by reshaping image and then executing 2D-based feature extraction methods. Specifically, we follow the same assumption as made in SSSL (a typical ESR method) that a small image region around an image pixel is smooth, and reshape each original image into a new matrix whose each column corresponds to a vectorized small image region, and then we extract features from the newly-formed matrix using any off-the-shelf 2D-based method which can take the relationship between pixels in the same row or column into account, such that the original spatial relationship within the neighboring region can be greatly retained. Since ISR does not impose constraint items, compared with ESR, ISR not only avoids the selection of the troublesome regularization parameter, but also greatly reduces computational cost. Experimental results on four face databases show that the proposed ISR can achieve competitive performance as SSSL but with lower computational cost.

1. Introduction

Face recognition [1–10], as one of the most important issues in computer vision and pattern recognition, has been advanced and widely studied over the past few decades because of its wide applications in security, human-machine communication, etc. Different from conventional image retrieval [11,12] or recognition [13] tasks, face recognition has its own challenges and attracted extensive research efforts. Among the existing face recognition methods, subspace learning method [1–10] is one of the most successful and well-studied techniques. In implementing the subspace learning methods, one need to first convert a two-dimensional (2D) face image of size $m \times n$ into a one-dimensional (1D) vector of length mn, i.e., representing each face image as a corresponding point in the high-dimensional vector space, and then apply features extraction for face recognition. However, such a vector conversion often suffers from two main problems: 1) small sample size (SSS) problem which leads to over-fitting in classification, and likewise makes the subspace learning methods (e.g., LDA, LPP, NPE, etc.) difficult to discover the real intrinsic discriminant or geometrical structures [14]; and 2) it breaks the natural spatial structure of images and thus makes the concatenated vector losing spatial relationship between pixels. In this paper, we attempt to address such two problems.

In order to retain the spatial relationship between image pixels as much as possible and at the same time avoid SSS problem, researchers have paid a lot of attention on original matrix or 2D representation of face images and developed corresponding 2D methods by directly operating on face matrices, for the vector-based subspace dimensionality reduction methods especially for PCA and LDA. The 2D versions of PCA include two-dimensional

* Corresponding author. Tel.: +86 2584892956.
E-mail addresses: lianyi_1999@nuaa.edu.cn (Y. Zhu), s.chen@nuaa.edu.cn (S. Chen).

http://dx.doi.org/10.1016/j.neucom.2015.09.028
0925-2312/

Input: Training face images $A = [A_1, A_2, \dots, A_N]$ where the size of each A_i is $m \times n$; the size of spatial window $p \times q$; the test image P.

Output: Class label of P.

Step 1: Image reshaping.

For $i = 1 : N$, % for each training face image

Partitioning A_i into some smaller patches with size of $p \times q$, then reshaping each patch into a column vector, after that collecting all column vectors to form a new reshaped matrix $A_i^{'}$ for A_i.

Step 2: Feature Extraction using 2D-based methods.

Solving the projection matrix U (or U and V) by applying 2D-based methods on the new reshaped matrix set $A^{'} = [A_1^{'}, A_2^{'}, \dots, A_N^{'}]$.

Computing features set $Y = [Y_1, Y_2, \dots, Y_N]$ for training image set by projecting each $A_i^{'} (i = 1,2, \dots, N)$ into U (or U and V) using $Y_i = U^T A_i$(or $Y_i = U^T A_i^{'} V$).

Step 3: Recognition.

1) Reshaping P like Step 1 and getting reshaped matrix P';

2) Extracting features F by using $U^T P^{'}$ (or$U^T P^{'} V$);

3) Computing similarities between $F$and each Y_i $(i = 1,2, \dots, N)$;

4) Outputting the class label of such training sample which has the maximum similarity with$F$.

Fig. 1. Pseudocode of the proposed ISR.

principal component analysis (2DPCA) [15], generalized low rank approximations of matrices (GLRAM) [16], etc. While the 2D versions of LDA include single-side 2D-LDA [17] and bi-side 2DLDA [18], etc. Among these methods, 2DPCA [15] and 2D-LDA [17] extract features only along the row (or the column) direction of image matrices, while GLRAM and 2DLDA can extract features along both the row and column directions of image matrices. Subsequently some researchers have further extended the subspace dimensionality reduction methods to higher order (HO) tensor data [19–21]. Until now, almost all existing vector-based subspace methods have been successively extended to their corresponding 2D or HO counterparts. Compared with the vector-based approaches, 2D-based approaches have three main advantages: 1) they can naturally and effectively elude SSS problem due to far lower dimensionality of the scatter matrix directly defined on images themselves, thus effortlessly avoiding singularity problem of the original within-class scatter matrix; 2) they partially utilize the spatial relationship among pixels in the same whole row (column); 3) they dramatically reduce the computational complexity in feature extraction due to small-size scatter matrix involved in dimensionality reduction. Extensive experimental results have shown the superiority of these 2D-based approaches to their corresponding vector counterparts. However, a recent research [14] pointed out that 2D-based approaches actually consider relationship between pixels only in the whole row (column) but fail to capitalize on the spatial information of the whole images such that the embedding functions of 2D-based approaches will still be spatially rough or not smooth enough.

To learn a spatially smooth subspace, recently, Cai et al. re-paid their attention to the original vector representation of face image and proposed a spatial regularization method called Spatially Smooth Subspace Learning (SSSL) [14], whose idea is quite general and thus suitable for almost all existing vector-based subspace methods. As a result, as expected, SSSL indeed achieves better recognition performance on benchmarks than the corresponding vector-based and 2D-based counterparts. In its implementation, the projection vectors are enforced to be spatially smooth by explicitly introducing a regularization term reflecting spatial relationship between pixels to the discriminant objectives of vector versions such as LDA. Following SSSL method, several variants [22,23] have been developed. In [22], Hou et al. proposed a orthogonal smooth subspace learning method (OSSL) by constraining the transformation vectors to be orthogonal and spatially smooth simultaneously; In [23], Zuo et al. improved SSSL by using LoG and DOG penalties as spatial regularization to replace the Laplacian penalty. Since these methods take into account the spatial relationship between image pixels by explicitly smoothing projection vectors of face space, they likewise significantly outperform their corresponding vector-based subspace learning methods without such regularization and 2D-based versions. However, these explicit spatial regularization (ESR) methods still have some disadvantages: first, compared with 2D-based methods, they have higher computational cost like traditional vector-based subspace methods; second, as aforementioned, they suffer from the relatively troublesome selection of key regularization factor which seriously influences the recognition performance and the optimal determination of the factor is still an open problem in machine learning, especially when the value of the factor is continuously changed; third, the size of image local region or window involved in spatial smoothness must be set to an odd value like 3×3, 5×5, or 7×7, etc., and each change of the spatial window size may make the selection of the regularization parameter be re-searched.

In this paper, we provide a new idea, coined as implicit spatial regularization (ISR), to retain the spatial relationship between image pixels and deal with SSS problem simultaneously for face recognition. Different from explicit spatial regularization (ESR), ISR is a direct realization for spatial relationship by just reshaping an original image matrix into another matrix but not need to introduce any explicit regularization term. Specifically, we use the prior knowledge that a small image region (hereafter it is called as spatial window) around a pixel is generally smooth, and reshape an original face image (denoted as a matrix) into a new alternative matrix by vectorizing each small image region into column vector, then we perform feature extraction on newly-formed matrix with the help

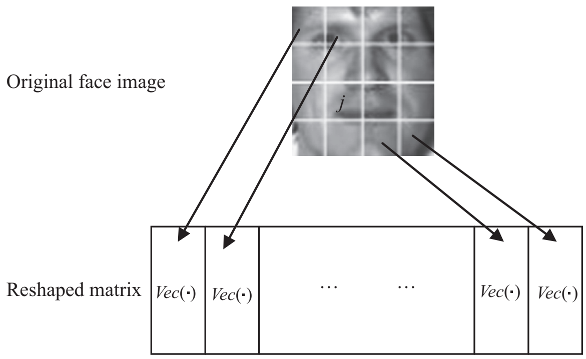

Fig. 2. An illustration of image reshaping in non-overlapping way, where Vec(•) is a simple vectorization operation.

of any off-the-shelf 2D-based method. Since a spatial window is reshaped into the same column, it can be desirable that the spatial structure within a spatial window can be obtained by using 2D-based methods. Compared with SSSL, our method not only avoids the selection of the troublesome regularization parameters, greatly reduces computational cost inherited from the 2D-based methods, but also the size of spatial window can be arbitrarily set according to the size of the whole face image. Compared with 2D-based methods, our method considers the spatial relationship of pixels within a small spatial region, rather than within a global row (or column). As a result, more spatial relationship can be used.

Note that, although our method does not explicitly introduce spatial regularization term to constrain spatial smoothness between neighboring pixels, it is implicitly accomplished by reshaping the image and then executing any off-the-shelf 2D-based method. Hence, in some loose sense, our method plays the role of spatial regularization, for which we coin it as implicit spatial regularization (ISR).

To evaluate efficacy of ISR, we compare ISR with SSSL on four face databases (Yale, ORL, Extended YALE B and CMU PIE), and the results showed that the proposed ISR-motivated methods achieve competitive performances against SSSL with lower computational cost. In addition, we also analyze the influence of the size of spatial window on performance of ISR.

The remaining parts of this paper are organized as follows. In Section 2, a brief review about SSSL and 2D-based subspace feature extraction methods is given. In Section 3, implicit spatial regularization (ISR) is formulated in detail. In Section 4, experimental comparisons carried out on four face databases are reported. Finally, a conclusion is drawn in Section 5.

2. Brief review of SSSL and 2D-based feature extraction methods

Let $A=[A_1, A_2, \ldots, A_N]$ be the training face image set and its corresponding vectorized training set be $X=[X_1, X_2, \ldots, X_N]$, where N is the number of training samples, $A_i (i=1,2,\ldots,N)$ is the i-th face image with the size of $m \times n$, and X_i is the vector representation of A_i. Also, let G be a graph constructed from A (or X), W and L be the edge weight matrix and graph Laplacian matrices associated with the G, respectively. With these definitions, we below briefly review Spatially Smooth Subspace Learning (SSSL) and related 2D-based subspace methods.

2.1. Spatially Smooth Subspace Learning (SSSL)

Spatially Smooth Subspace Learning (SSSL) [14] is a typical explicit vector-based spatial regularization method which is suitable for almost all existing subspace methods. Its starting point is

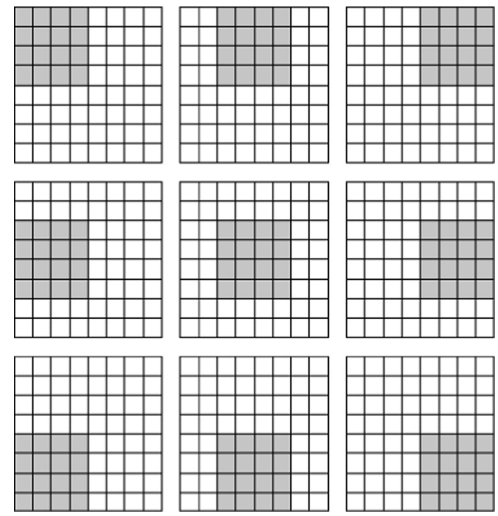

Fig. 3. An illustration of image reshaping in overlapping way.

to encourage projection vectors to be spatially smooth by using discrete Laplacian smooth operator on the projection vectors. As a result, SSSL can achieve better performance than the corresponding subspace learning methods without such a spatial smoothness constraint [14]. Mathematically, the objective function of SSSL method can be formulated as

$$\arg\max_{a} \frac{a^T XWXa}{(1-\beta)a^T XLX^T a + \beta J(a)} \tag{1}$$

where a is the projection vector to be solved, β is a regularization factor, and J is a discrete Laplacian penalty function defined as:

$$J(a) = a^T \Delta^T \Delta a \tag{2}$$

where Δ is a discrete approximation for two-dimensional Laplacian matrix with the size of $mn \times mn$:

$$\Delta = D_1 \otimes I_2 + I_1 \otimes D_2 \tag{3}$$

where I_1 and I_2 are $m \times m$ and $n \times n$ identity matrices, respectively, $\otimes$ denotes the Kronecker product operator, and D_1 (D_2) is a modified Neuman horizontal (vertical) direction discretization operator defined by

$$D_j = \frac{1}{h_j^2} \begin{pmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ \cdot & \cdot & \cdot \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{pmatrix} \tag{4}$$

where h_1 (h_2) is the sample width along the horizontal (vertical) direction. For more details, please refer to [7].

According to the above description, we can summarize that 1) SSSL fixes the size of spatial (smooth) window to 3×3 and does not further explore the influence of the size of spatial window on classification performance. Although the size of such spatial window can take other numbers, it is limited to some odd values such as 3×3, 5×5, or 7×7, etc. due to the requirement of the used smooth operators; 2) as a vector-based method, SSSL suffers from high computational cost; and 3) as an explicit spatial regularization method, SSSL needs to expensively tune the regularization factor for model selection, especially when its value continuously varies, thus screening out optimal factor will be more difficult.

Input: Training face images $A' = [A_1', A_2', \dots, A_N']$, weight matrix W'; the initialization V_0; number of iteration J.

Output: $U$and V

Step 1: Computing diagonal matrix $D_{i,i} = \sum_{i \neq j} W_{i,j}'$

Step 2: $V \leftarrow V_0$

Step 3: For $j = 1 : J$,

Computing the first u eigenvectors $\phi_1, \dots, \phi_u$ of $\sum_{i,j} W_{i,j}' A_i' V V^T A_j'^T \phi = \lambda \sum_i D_{i,i} A_i' V V^T A_j'^T \phi$

$U \leftarrow \{\phi_1, \dots, \phi_u\}$

Computing the first v eigenvectors $\varphi_1, \dots, \varphi_v$ of $\sum_{i,j} W_{i,j}' A_i'^T U U^T A_j' \varphi = \lambda \sum_i D_{i,i} A_i'^T U U^T A_j' \varphi$

$V \leftarrow \{\varphi_1, \dots, \varphi_v\}$

end

Step 4: return (U, V).

Fig. 4. Pseudocode of solving U and V.

2.2. 2D-based feature extraction methods

In this paper, since our focus is not on improving existing 2D-based methods but applying them as off-the-shelf methods, we just give a brief review about 2D-based methods.

2D-based feature extraction methods are designed especially for manipulating 2D images and have been effectively applied in face recognition. In their implementation, each face image is directly represented as a matrix on which 2D-based feature extraction methods are performed. These 2D-based methods for feature extraction can be classified into two categories: one class is single-side method which extracts features only along the row (or column) direction, and the other one is called bi-side method which simultaneously extracts features along both the row and column directions.

The aim of single-side 2D methods is to find a column projection matrix U or a row projection matrix V by optimizing some specific criteria such as the reconstruction error [15,16] and linear discriminant criteria [18], and then extract features $U^T A_i$ (or $A_i V$) from the given face image A_i. An attractive property of the single-side methods is that it can yield an analytic solution for U (or V) through solving a generalized eigen-equation.

Different from the single-side 2D methods, the bi-side 2D methods aim to find a column projection matrix U and a row projection matrix V simultaneously. The optimization criteria with respect to both U and V are not jointly convex, so one usually adopt an alternating iteration strategy to solve U and V. More specifically, for a fixed V, the optimal U is first computed by solving an optimization problem similar to single-side methods. With the computed U, V is updated by solving the other optimization problem. Such an alternative optimization process is repeated until convergence of the value of the objective.

3. Implicit spatial regularization (ISR)

In this section, we provide a nominal yet simple spatial regularization method to avoid the loss of spatial information between image pixels and deal with SSS problem, simultaneously. The brief description of our proposed ISR algorithm is summarized in Fig. 1.

3.1. Related works on reshaping and motivation

Reshaping images has been used in face recognition [24–27]. For example, in our previously proposed methods MatPCA and MatLDA [24], we first converted a general vector pattern into a corresponding matrix pattern and then performed 2DPCA [15] and 2D-LDA [17] on the converted matrix to extract features. The motivation of doing so is to expect introducing some structural information potentially hidden in data itself. In [25], Xu et al. rearranged pixels of a given image to make correlations along certain tensor dimensions maximal with an aim of removing as much information redundancy as possible, and applied existing tensor-based algorithms to reduce dimensionality. In [26], Chen et al. proposed a Glocal image representation with a goal to introduce two structurally-meaningful vector spaces to respectively describe the global and the local image properties for learning image metric. In [28], Kumar et al. transformed the original image into new representation so that the convolution operation can always be represented as a linear operation. All the above methods have shown their effectiveness to different degrees in face recognition.

In this paper, we also adopt the reshaping trick for face recognition. Our motivation to reshape image comes from the following observation: as analyzed in SSSL that 2D-based methods can obtain the spatial relationship within the whole row or column, however, the whole row or column space of image is spatially rough or under-smoothness, as a result, they fail to fully explore the spatial information of images. Since keeping smoothness of spatial window is effective to improve the generalization performance of subspace methods, we consider reshaping original face image to increase smoothness of column vector and further obtain the spatial relationship between pixels within the same column vector by using tensor methods. That is, we reshape a given face image (as a matrix) into an alternative matrix whose each column corresponds to a vectorized spatial window, as a result, the column space of the new matrix becomes enough smooth. When 2D-based feature extraction method is performed on the reshaped matrix, the spatial relationship within spatial window will be considered.

3.2. Reshaping

Reshaping can usually be realized in non-overlapping and overlapping ways. In non-overlapping way, we first divide a given face image size of $m \times n$ into a set of equally-sized $(p \times q)$ spatial windows (forming $K = (m \times n)/(p \times q)$ spatial windows) and then convert each one into corresponding pq-dimensional column vector to form a new matrix with size of $pq \times K$. This reshaping process is simply illustrated in Fig. 2. Obviously, after reshaping, the columns (corresponding spatial windows) of the new matrix are usually smooth and characterize the local configurations while the rows characterize the global configurations, thus when bi-side 2D-based methods are used to extract features, the local spatial relationship within each spatial window (along the column direction) and global features (along the row direction) can be retained to great degree.

Such a non-overlapping partition breaks likely the spatial relationship between neighboring windows due to non-smooth sliding between them and thus leads to ignorance of the relations to some degree. On the other hand, the overlapping partition way can usually mitigate such issue due to the connection of adjacent spatial windows and the combination of different information between them. Fig. 3 illustrates the construction process of spatial windows in an overlapping way for image of size 8×8 and spatial window of size 4×4, where the gray grid stands for spatial window. As a result,9 spatial windows are generated and the size of newly-formed matrix is 16×9.

3.3. Feature extraction using single- and bi-side 2D methods

After reshaping image, we can perform any off-the-shelf 2D-based method on reshaped images to extract features. Let $A' = [A'_1, A'_2 \ldots A'_N]$ be a set of reshaped training matrix belonging to C classes and the i-th class C_i has n_i images. Also, let $G' = < A', W' >$ be an undirected weighted graph with vertex set being A', the weight matrix being W' whose entry $W'_{i,j}$ denotes the similarity between vertexes A'_i and A'_j and D be a diagonal matrix where $D_{i,i} = \sum_{i \neq j} W'_{i,j}$. In the following, we will briefly describe these 2D-based feature extraction methods involved.

In the paper, we use two families' algorithms, LDA family and LPP family, in order to obtain different projection matrices. The aim of LDA is to find a set of projection directions by maximizing the between-class scatter matrix and minimizing the within-class scatter matrix to achieve as large between-class separation as possible in the reduced subspace; while the aim of LPP family is to find local and intrinsic low-dimensional sub-manifold structures hidden in high-dimensional space. Although there are different motivations for LDA and LPP, they can be nicely interpreted in a general graph embedding framework. If $Y = [Y_1, Y_2, \ldots, Y_N]$ is the extracted feature set, the objective function based on graph embedding framework can be formulated as follows:

$$\min \sum_{i,j} \left\| Y_i - Y_j \right\|^2 W'_{i,j} \tag{5}$$

LDA and LPP can be interpreted in this graph framework with different weight matrix W'. For LDA, the weight can be defined as follows:

$$W_{i,j}^{LDA} = \begin{cases} \frac{1}{n_t}, \text{ if } A_i' \text{ and } A_j' \text{ belong to } i-\text{th class} \\ 0, \text{ otherwise} \end{cases} \tag{6}$$

As for LPP, the weight is

$$W_{i,j}^{LPP} = \begin{cases} e^{-\frac{\|A_i' - A_j'\|^2}{2\sigma^2}}, \text{ if } A_i' \in N_k\left(A_j'\right) \text{ or } A_j' \in N_k\left(A_i'\right) \\ 0, \text{ otherwise} \end{cases} \tag{7}$$

where $N_k\left(A_i'\right)$ denotes the set of k nearest neighbors of A_i', and σ is a parameter.

3.3.1. Feature extraction using single-side 2D methods

The aim of single-side 2D methods is to find the projection matrix U (or V) then project any given matrix A'_i into U (or V) to obtain corresponding features using $Y_i = U^T A_i'$ (or$Y_i = A_i'V$). Since one of aims of ISR is trying to use as much spatial structure among neighboring pixels as possible, we prefer to extract features from the reshaped images along column direction, i.e., extracting features by $Y_i = U^T A_i'$.

If replacing Y_i using $U^T A'_i$, we have

$$\min \sum_{i,j} \left\| Y_i - Y_j \right\|^2 W'_{i,j} = \min \sum_{i,j} \left\| U^T A'_i - U^T A'_j \right\|^2 W'_{i,j} \tag{8}$$

With some simple algebraic formulations, minimization problem in Eq. (8) can be rewritten as

$$\max U^T \sum_{i,j} W'_{i,j} A'_i A_j'^T) U \tag{9}$$

$$\text{s.t.} U^T \sum_{i,i} (D_{i,i} A_i' A_i'^T) U = 1$$

It is easy to see that the optimal U can be obtained by solving the maximum eigen-value problem:

$$\sum_{i,j} (W'_{i,j} A'_i A_j'^T) U = \lambda \sum_i (D A'_i A_i'^T) U \tag{10}$$

Using different weight matrix $W' \left(W^{LDA} \text{ or } W^{LPP} \right)$, we can obtain the single-side 2D projection matrix U (U_{LDA} or U_{LPP}) for 2D-LDA [17] or 2DLPP [29]. Once we obtain projection matrix U, we can extract features from A_i' using $U^T A_i'$.

3.3.2. Feature extraction using bi-side 2D methods

Different from the single-side 2D methods, the bi-side 2D methods extract simultaneously features from both the row and column directions. Let U and V be the column and the row

Table 1
Comparison on Yale database using single-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Single-side method	*G4*	*G5*	*G6*	*G7*
ISR-LDA	75.2 ± 3.2	81.3 ± 3.9	85.0 ± 3.9	86.0 ± 3.6
ISR-LDA-overlapping	76.2 ± 4.1	**82.3 ± 3.8**	**86.7 ± 3.9**	**86.9 ± 2.6**
S-LDA	**77.8 ± 3.0**	81.7 ± 3.2	82.7 ± 4.4	84.2 ± 3.5
ISR-LPP	75.5 ± 3.3	81.3 ± 3.6	85.7 ± 3.8	86.3 ± 3.2
ISR-LPP-overlapping	**76.8 ± 3.5**	**82.5 ± 3.2**	**86.8 ± 4.0**	**87.4 ± 3.2**
S-LPP	76.0 ± 3.4	81.4 ± 2.9	82.5 ± 4.5	84.0 ± 4.3

Table 2
Comparison on Yale database using bi-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Bi-side method	*G4*	*G5*	*G6*	*G7*
ISR-LDA	79.8 ± 4.2	83.4 ± 3.6	86.2 ± 3.3	87.9 ± 2.6
ISR-LDA-overlapping	**80.2 ± 3.1**	**84.4 ± 3.2**	**87.9 ± 4.7**	**88.7 ± 4.0**
S-LDA	77.8 ± 3.0	81.7 ± 3.2	82.7 ± 4.4	84.2 ± 3.5
ISR-LPP	79.7 ± 4.1	83.8 ± 3.2	86.5 ± 3.7	87.9 ± 3.5
ISR-LPP-overlapping	**80.1 ± 3.0**	**84.6 ± 2.8**	**87.3 ± 4.5**	**89.1 ± 3.5**
S-LPP	76.0 ± 3.4	81.4 ± 2.9	82.5 ± 4.5	84.0 ± 4.3

Fig. 5. Sample images from the ORL database.

projection matrices respectively acting on the left- and right-side of a matrix whose dimensionality is to be reduced, i.e.,$Y_i = U^T A'_i V$. Correspondingly, the objective function in Eq. (5) can be rewritten as:

$$\min\sum_{i,j}\|Y_i - Y_j\|^2 W'_{i,j} = \min\sum_{i,j}\|U^T A'_i V - U^T A'_j V\|^2 W'_{i,j} \tag{11}$$

With some simple algebraic formulations, minimization problem in Eq. (11) can be rewritten as Eqs. (12) or (13)

$$\max U^T \sum_{i,j}(W'_{i,j} A'_i V V^T A'^T_j) U \tag{12}$$

$$\text{s.t. } U^T \sum_{i,i}(D_{i,i} A'_i V V^T A'^T_i) U = 1$$

$$\max V^T \sum_{i,j} W'_{i,j} A'^T_i U U^T A'_j) V \tag{13}$$

$$\text{s.t. } V^T \sum_{i,i}(D_{i,i} A'^T_i U U^T A'_i) V = 1$$

In the bi-side 2D methods, since U and V are simultaneously involved and the optimization criteria are not jointly convex w.r.t. U and V but biconvex, i.e., fixing either leads theoptimization criteria to be convex w.r.t. the other, thus we have to obtain them by resorting to the alternating iteration optimization strategy [19]. Concretely, we first fix V to optimize the objective function Eq. (12) to obtain U as done in the single-side method, and then fix the last-step U to optimize the objective function Eq. (13) to obtain V. Such an optimization process is alternatively repeated until the convergence of the value of the objective. The pseudocode of solving U and V for bi-side 2D methods is shown in Fig. 4.

Similarly, with different weight matrix $W'(W^{LDA}$ or $W^{LPP})$, we can obtain the bi-side 2D projection matrix U ($U_{LDA} or U_{LPP}$) and V (V_{LDA} or V_{LPP}) for 2DLDA [18] or TSA [19]. At the same time, we can extract features from A'_i by using $U^T A'_i V$.

3.4. Recognition

Let P_1 and P_1 be two given face images and their corresponding extracted features with the vector representation are denoted by $Z_1 = [z_1^1, z_1^2, \ldots, z_1^d]$ and $Z_2 = [z_2^1, z_2^2, \ldots, z_2^d]$, respectively. Then the similarity between them, $d(Z_1, Z_2)$ can be defined as

$$\text{similarity}(Z_1, Z_2) = 1 - \sum_{k=1}^{d} \left\| z_1^k - z_2^k \right\|^2 \tag{14}$$

where $\|\bullet\|$ denotes the Euclidean distance.

When a known face image P is given, we first convert it into a new matrix P' by reshaping described in Section 3.2, and then extract features F using $U^T P'$ (or $U^T P' V$) in terms of single-side (or bi-side) method. Let $Y_1, Y_2, \ldots, Y_N$ be the features extracted from all of training image. If $\text{similarity}(Y_t, F) = \max(\text{similarity}(Y_i, F))$ for all $i = 1, 2, \ldots, N$, and $Y_t \in C_o$, a final decision is made as $P \in C_o$.

4. Experiments and analysis

4.1. Experimental settings

In order to evaluate the recognition performance of our method ISR, we carry out some experiments on four benchmark face databases: the Yale database, the Olivetti Research Laboratory (ORL) database, the Extended Yale B database and the CMU PIE

Table 3
Comparison on ORL database using single-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Single-side method	*G4*	*G5*	*G6*	*G7*
ISR-LDA	93.9 ± 1.7	96.7 ± 1.3	97.6 ± 1.0	97.6 ± 1.5
ISR-LDA-overlapping	95.3 ± 1.4	**97.5** ± **1.5**	**98.3** ± **0.9**	**98.7** ± **1.1**
S-LDA	**95.8** ± **1.3**	97.2 ± 1.3	97.7 ± 1.2	98.1 ± 1.5
ISR-LPP	93.9 ± 1.8	96.7 ± 1.3	97.7 ± 1.1	97.6 ± 1.5
ISR-LPP-overlapping	95.4 ± 1.4	**97.5** ± **1.5**	**98.3** ± **0.9**	**98.6** ± **1.2**
S-LPP	**95.8** ± **1.3**	97.2 ± 1.3	97.2 + 1.1	97.4 ± 1.6

Table 4
Comparison on ORL database using bi-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Bi-side method	*G4*	*G5*	*G6*	*G7*
ISR-LDA	95.6 ± 1.6	97.4 ± 1.4	98.2 ± 0.8	98.3 ± 1.4
ISR-LDA-overlapping	**96.4** ± **1.1**	**98.0** ± **1.2**	**98.7** ± **0.9**	**98.8** ± **1.1**
S-LDA	95.8 ± 1.3	97.2 ± 1.3	97.7 ± 1.2	98.1 ± 1.5
ISR-LPP	95.5 ± 1.5	97.4 ± 1.3	98.2 ± 0.9	98.3 ± 1.4
ISR-LPP-overlapping	**96.4** ± **1.1**	**98.0** ± **1.3**	**98.7** ± **0.9**	**98.8** ± **1.1**
S-LPP	95.8 ± 1.3	97.2 ± 1.3	97.2 ± 1.1	97.4 ± 1.6

Table 5
Comparison on Yale B database using single-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Single-side method	*G10*	*G20*	*G30*	*G40*
ISR-LDA	86.7 ± 1.3	93.3 ± 0.6	95.7 ± 0.6	96.4 ± 0.4
ISR-LDA-overlapping	87.7 ± 1.2	**93.8** ± **0.5**	**96.6** ± **0.7**	**97.1** ± **0.4**
S-LDA	**88.2** ± **1.3**	92.9 ± 1.0	94.5 ± 0.6	95.2 ± 0.8
ISR-LPP	86.8 + 1.3	93.5 ± 0.7	95.6 ± 0.6	96.4 ± 0.4
ISR-LPP-overlapping	87.3 ± 1.4	**94.0** ± **0.7**	**96.2** ± **0.7**	**97.2** ± **0.5**
S-LPP	**87.8** ± **1.3**	92.8 ± 0.5	94.5 ± 0.6	94.9 ± 0.9

Table 6
Comparison on Yale B database using bi-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Bi-side method	*G10*	*G20*	*G30*	*G40*
ISR-LDA	88.2 ± 1.2	93.9 ± 0.6	95.8 ± 0.6	96.5 ± 0.6
ISR-LDA-Overlapping	**90.6** ± **1.2**	**94.6** ± **0.5**	**96.1** ± **0.6**	**97.0** ± **0.6**
S-LDA	88.2 ± 1.3	92.9 ± 1.0	94.5 ± 0.6	95.2 ± 0.8
ISR-LPP	88.0 ± 1.3	93.8 ± 0.6	95.8 ± 0.6	96.5 ± 0.6
ISR-LPP-Overlapping	**89.2** ± **1.2**	**94.4** ± **0.5**	**96.2** ± **0.7**	**96.8** ± **0.7**
S-LPP	87.8 ± 1.3	92.8 ± 0.5	94.5 ± 0.6	94.9 ± 0.9

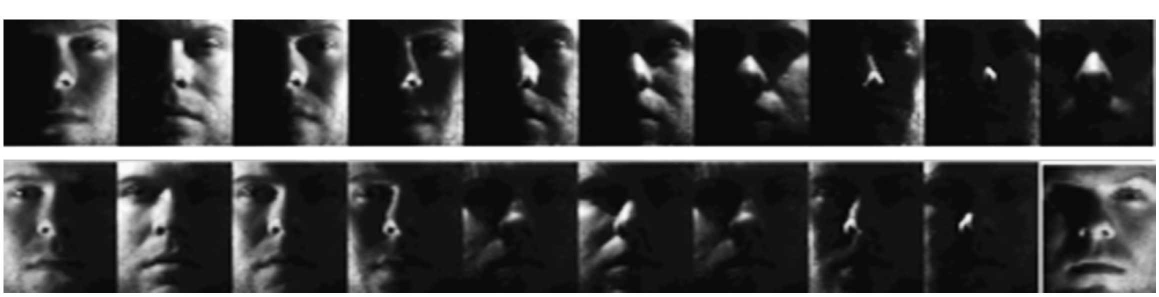

Fig. 6. Sample images from the extended Yale B database.

Fig. 7. Sample images from CMU PIE database.

Table 7
Comparison on CMU PIE database using single-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Single-side method	*G*5	*G*10	*G*20	*G*30
ISR-LDA	70.74 ± 1.54	84.18 ± 0.79	91.62 ± 0.49	94.66 ± 0.41
ISR-LDA-Overlapping	**75.60 ± 1.41**	**86.75 ± 0.82**	**92.71 ± 0.50**	**95.12 ± 0.37**
S-LDA	66.21 ± 1.49	81.61 ± 0.81	90.86 ± 0.56	94.00 ± 0.34
ISR-LPP	70.73 ± 1.35	84.21 ± 0.85	91.55 ± 0.54	94.64 ± 0.42
ISR-LPP-Overlapping	**75.14 ± 1.39**	**86.63 ± 0.77**	**92.67 ± 0.52**	**94.94 ± 0.36**
S-LPP	63.01 ± 1.72	77.79 ± 0.81	87.83 ± 0.69	91.70 ± 0.40

Table 8
Comparison on CMU PIE database using bi-side methods (mean ± std-dev%) (the best performance in each case has been bolded).

Bi-side method	*G*5	*G*10	*G*20	*G*30
ISR-LDA	74.34 ± 1.22	86.80 ± 0.76	92.92 ± 0.48	94.57 ± 0.37
ISR-LDA-overlapping	**77.92 ± 1.16**	**89.39 ± 0.67**	**94.91 ± 0.44**	**96.15 ± 0.31**
S-LDA	66.21 ± 1.49	81.61 ± 0.81	90.86 ± 0.56	94.00 ± 0.34
ISR-LPP	74.14 ± 1.34	86.65 ± 0.85	92.62 ± 0.58	94.60 ± 0.29
ISR-LPP-overlapping	**77.31 ± 1.41**	**89.33 ± 0.69**	**94.83 ± 0.50**	**96.05 ± 0.32**
S-LPP	63.01 ± 1.72	77.79 ± 0.81	87.83 ± 0.69	91.70 ± 0.40

database. Considering the specific characteristics of these four databases, the Yale database is employed to test the performance of ISR under various facial expressions and slight lighting conditions. The ORL is used to test the robustness of ISR to slight pose variation. The Extended Yale B database is utilized to evaluate the performance under severe variations of illumination. The CMU PIE database is utilized to examine the performance under multiple variations, i.e., slight lighting conditions, pose variation and expression variation. All used images were provided by Cai (available at: http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html) by manually cropping and resizing original images to the size of 32×32, and we did not do any extra processing. In [14], the authors compared SSSL with ordinary vector-based methods and their 2D extensions, and their experimental results show that SSSL significantly outperforms those compared subspace methods, so in this paper, we only compare our ISR (ISR-LDA and ISR-LPP) with SSSL (S-LDA and S-LPP) for each LDA and LPP families. In our experiments, we use non-overlapping and overlapping ways to partition an original image into some spatial windows. For non-overlapping way, we uniformly set the size of spatial window to 3×3 and adopt nearest neighbor interpolation method to resize face image to 33×33 in order to partition each face image into a set of equally-sized spatial window. In experiments, we split each database into two subsets: the training set and the test set, and use Gl to denote that l images of each individual are used for training and the remaining images for test. We run 20 times for each method and report the average recognition accuracies and standard deviations.

4.2. Experiments on the Yale face database

The Yale face database consists of 165 face images of 15 individuals, each providing 11 different images. The images are in upright, frontal position under various facial expressions and lighting conditions. Tables 1 and 2 list the average recognition accuracies and standard deviations on Yale database using single-side and bi-side 2D methods, respectively. From Table 1, we can get the following observations: 1) in both overlapping and non-overlapping ways, the ISR exhibits its efficacy and competitiveness against explicit regularization method SSSL (S-LDA and S-LPP); 2) the overlapping partition way for spatial window is more effective in performance than the non-overlapping one, which can attribute to the fact that the overlapping partition can connect the adjacent spatial windows and combine the different information in each spatial window, thus more structural information can be utilized; 3) although the single-side non-overlapping ISR is inferior to the SSSL when the number of training set is small, with the increase of the number of training samples, the recognition accuracies of ISR is superior to the SSSL.

Similar conclusions can be drawn from Table 2. Using the bi-side 2D methods, ISR in both overlapping and non-overlapping ways is superior to SSSL in performance. By a further comparison between the results in Tables 1 and 2, we can find that ISR based on bi-side methods achieve higher recognition accuracies than ISR based on single-side methods, which owns to the fact that bi-side 2D methods consider both the row and the column structural information of face image simultaneously, while single-side methods only consider the column-alone information.

4.3. Experiments on the ORL face database

The ORL database contains images from 40 subjects, with 10 different images for each subject. For some subjects, the images were taken at different sessions. There are variations in facial expressions (open or closed eyes, smiling or non-smiling), facial details (glasses or no glasses) and scale (up to about 10%). Moreover, the images were taken with a tolerance for tilting and rotation of the face of up to 20°. Fig. 5 shows all samples of one person from the ORL database and the averaged results over 20 runs are reported in Tables 3 and 4. The results show that: 1) compared with SSSL, ISR is competitive in face recognition performance; and more importantly 2) ISR is not sensitive to image slight variations in pose angle and mis-alignment.

4.4. Experiments on the Extended Yale B face database

The Extended Yale B face database contains 21,888 single light source images of 38 subjects captured under 576 viewing conditions (9×64 illumination conditions). In this paper, we only use a subset provided by Lee et al. [30], which only contains these images with the frontal pose for each individual, including 2432 images from 38 subjects and all of face images with the same subject have minor differences beside of the lighting condition. Fig. 6 shows some cropped images of one person from Extended Yale B database.

Tables 5 and 6 list the recognition performance for different methods using single-side and bi-side 2D methods, respectively. From these two tables, we not only draw the similar conclusions with in Yale and ORL, that is, both ISR-LDA and ISR-LPP gain the competitive recognition performance against S-LDA and S-LPP. More importantly, our ISR is very robust to severe lighting variation. The main reason is that some of the local facial features of an individual do not vary when the lighting direction (or pose) varies, hence the features from the regions not affected by illumination (or pose) will closely match with the features of the same individuals

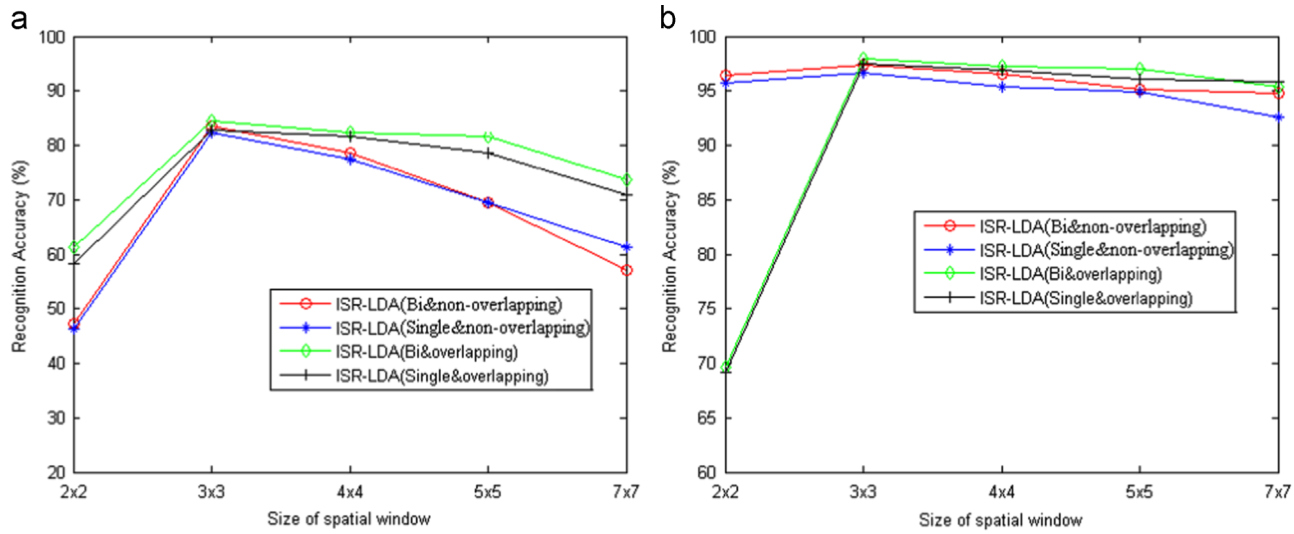

Fig. 8. The influence of the size of spatial window on performance of ISR-LDA: (a) Yale (b) ORL.

Table 9
The influence of size of spatial window on S-LDA for face image with size of 32×32 (%).

Dataset	Size of window 3×3	5×5	7×7
Yale	81.7	75.8	69.8
ORL	97.2	95.4	87.3

face regions under normal conditions. In our method, each column vector of reshaped matrix corresponds to a local region and local features can be extracted from these column vectors by using the 2D methods, so better recognition accuracy with high robustness can be achieved.

4.5. Experiments on CMU PIE database

The CMU PIE face database contains 41,368 images of 68 people, each person under 13 different poses, 43 different illumination conditions, and with 4 different expressions. In experiments, the used database only contains five near frontal poses (C05, C07, C09, C27, C29), so there are 170 images for each individual. Fig. 7 shows some images of one person from the PIE database. We report the average recognition performance and standard deviations of different methods using single-side and bi-side 2D methods in Tables 7 and 8, respectively. From the two tables, we can observe that both ISR-LDA and ISR-LPP significantly outperform S-LDA and S-LPP, especially when the number of training samples per individual is smaller. When using 5 training samples per class, the maximum difference between SSSL and ISR is 8%, and the minimum difference is 4.5%. We think this can attribute to characteristics of CMU PIE database. As described above, CMU PIE database consists of some images with multiple variations, i.e., slight lighting conditions, pose variation and expression variation. When number of training sample is very small, it is very difficult to learn good regularization parameters for SSSL, so performance of SSSL is very poor. In contrast, our method can directly extract local information from the column vectors of the reshaped matrix, and the local information is not affected too much, so it can achieve the satisfying accuracies.

4.6. Parameter selection

The size of spatial window and the number of projection vectors are two key parameters involved in our proposed method ISR. In this subsection, we will take LDA as an example to study the influence of the two parameters on the recognition performance of ISR by using Yale and ORL databases with 5 training samples per class.

We first study the influence of the size of spatial window on ISR. In fact, the size of spatial window is an essential parameter in ISR. Intuitively, the smaller is the size; the less spatial information is obtained. One extreme case is that the size of spatial window is 1×1, i.e., only one-pixel size. On the contrary, the size should also not be too big. The bigger is the size; the rougher is the spatial window. Another extreme case is the size of spatial window is equal to that of the whole image. So we need make a compromise to select an appropriate size of spatial window. Along this line, we conduct experiments on the original images with the size of 32×32 and resize the face images by adopting the nearest neighbor interpolation method with the aim of partitioning the face images into a set of equally-sized spatial windows in either non-overlapping or overlapping way. The results with respect to different spatial window sizes are demonstrated in Fig. 8. These results verify our intuitional analysis, i.e., the size of spatial window should be neither too small nor too big. At the same time, we observe that ISR achieves the best recognition accuracies when the size of spatial window is selected as 3×3 (about 1/10 of the original image size) in both non-overlapping and overlapping ways. Similar conclusion has been drawn that it is most helpful to improve the recognition performance of sub-image methods when block (or window) size is about 1/10 of the original size [31].

From the Fig. 8, we also observe that there are different change trends against the size of spatial windows for Yale and ORL databases. For Yale database (shown in Fig. 8(a)), the variation of performance with respect to different window sizes is very big (the maximum difference is up to 20%); while for ORL database (shown in Fig. 8(b)), except for 2×2, the variation of performance with respect to different window sizes is small (the maximum difference is less than 5%). This difference of change trends can attribute to the different characteristics of Yale and ORL databases. As we know, the ORL database consists of some images with variations of scale, pose, rotation and tilting; while the Yale database only consists of some frontal face images with upright, frontal position under various facial expressions and lighting conditions.

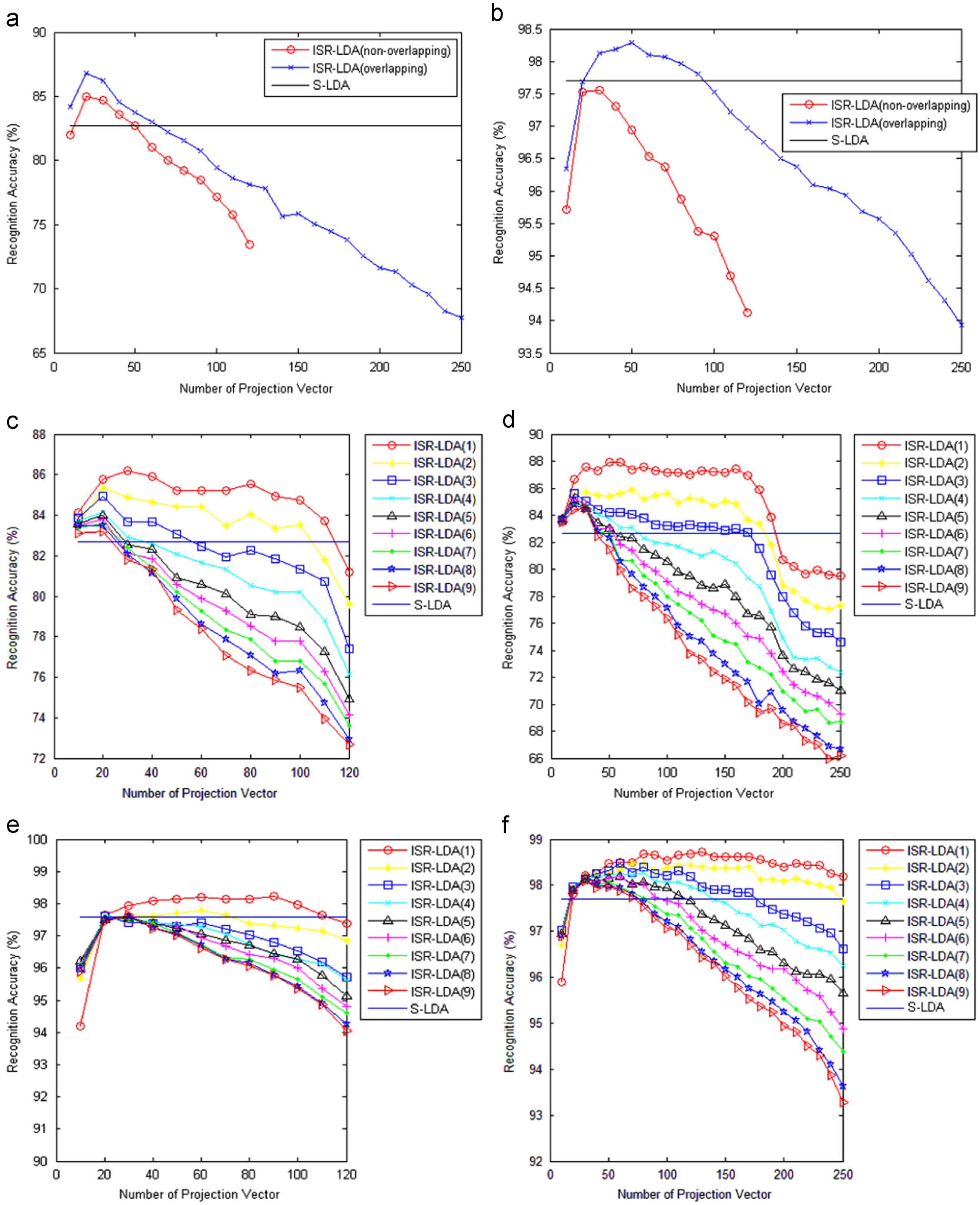

Fig. 9. Influence of the number of projection vectors on ISR-LDA: (a) single side for Yale; (b) single side for ORL; (c) non-overlapping and bi-side for Yale; (d) overlapping and bi-side for Yale; (e) non-overlapping and bi-side for ORL; (f) overlapping and bi-side for ORL.

In this situation, global features extracted from the whole image may play more important role than local features extracted from each spatial window for ORL database; while local features extracted from the spatial window may be more important than global features for Yale database. Since the column vectors and row vectors of reshaped matrix descript the local features (corresponding to the spatial window) and the global features, respectively, and different spatial window size will cause a strong change in the local information, while relatively small change in the global information. As a result, the recognition performance is relatively more stable on ORL database than on Yale database when using different size of spatial window.

Table 10
Time comparisons for S-LDA and ISR-LDA (in seconds).

	ISR-LDA (single-side, overlapping)	ISR-LDA (single-side, non-overlapping)	ISR-LDA (bi-side, overlapping)	ISR-LDA (bi-side, non-overlapping)	S-LDA
Time	15.07	5.80	118.79	26.17	38.74

In addition, for ORL database, we also observe that ISR in non-overlapping partition is much better than that in overlapping partition way when size of spatial window is 2×2. The main reason is that there exists much redundancy information between row vectors when partitioning spatial window with size of 2×2 in an overlapping way. For an image with size of 32×32, the size of reshaped matrix in overlapping way is $4 \times$ (31*31), so there is much redundancy information between row vectors of the reshaped matrix. Since global information described by each row vector is more important for ORL database and there exists much redundancy information between row vectors, the recognition accuracy is not satisfying. Note that, there also exists redundancy in Yale database, but the local information extracted from the column vector of reshaped matrix is more important than global information for this database, and the column vector has no much redundancy information, so ISR with overlapping partition achieved good performance.

As mentioned in previous introduction section, for SSSL, the authors just set the size of spatial (smooth) window to 3×3 and do not further analyze the influence of size setting on classification performance. In this work, we also simply examine the effect of spatial windows with different sizes on SSSL based on LDA (S-LDA).We conduct experiment on Yale and ORL databases with 5 training samples per class for the window sizes of 3×3, 5×5 and 7×7, respectively, and show their average recognition rates of 20 runs in Table 9. Interestingly, we witness likewise that S-LDA also gain the best recognition accuracies when the size of spatial window is set to about 1/10 of the size of original image.

Another parameter which needs to be tuned in ISR is the number of projection vectors. Since we use both single-side and bi-side 2D methods, in this section, we will discuss the impact of the number of projection vectors on both single-and bi-side ISR-LDA, respectively. We resize the image to 33×33 and set the size of spatial window to 3×3, as a result, the size of reshaped matrix is 9×121 and 9×256 when partitioning spatial window in non-overlapping and overlapping ways, respectively. For the bi-side method, we change the number of column projections from 1 to 9, and the number of row projections from 10 to 120 (in non-overlapping way) or 250 (in overlapping way) by interval 10; while for the single-side method, only the number of column projections is changed from 1 to 9. In our preliminary experiments, we find that the recognition accuracy of S-LDA increases monotonically with the number of projections, that is to say, S-LDA can gain the best recognition accuracy when the number of projection vectors is $C-1$ (where C is the number of classes). So we only show the best performance of S-LDA in order to emphasize the superiority of ISR-LDA. Fig. 9 shows the influence of the number of projection vectors on performance of ISR-LDA in both ORL and Yale databases, where n in ISR-LDA (n) denotes the number of column projections. From this figure, we can observe that 1) ISR-LDA, on the whole, is competitive to S-LDA over a large range of number of projection vectors especially the bi-side ISR-LDA in overlapping way; 2) unlike S-LDA, the recognition performance of ISR-LDA does not monotonically vary with the number of projection vectors but first increases then decreases, which shows that the number of projection dimensions should be neither too small nor too big. For single-side ISR-LDA, when the number of projections is between 20 and 40, and for bi-side ISR-LDA, when the number of column projections is 1 or 2 and the number of row projections is between 30 and 80, ISR-LDA can gain high recognition performance.

4.7. Time complexity for ISR-LDA and S-LDA

Efficiency is one of important factors to evaluate whether an algorithm is good or not. In this section, we will take LDA as an example to simply analyze the time complexities of SSSL and ISR. Let the size of face image be $m \times n$, the number of training samples be N and the size of spatial window be $p \times q$, then the time complexities of S-LDA and ISR-LDA are $O(m^3n^3)$ and $O(p^3q^3)$, respectively. Since p and q are usually far less than m and n, respectively, so ISR-LDA has less time complexity than S-LDA. Meanwhile, we list in Table 10 the running time-consuming for S-LDA and ISR-LDAs (single- and bi-side) on Yale database with 5 images for training per individual. As can be seen from this table, the ISR-LDAs (but the bi-side ISR-LDA in overlapping way) is much faster than S-LDA. All the algorithms are implemented in Matlab7.0 and run on an Intel Core 2 2.0 GHz PC with 2 GB memory.

5. Conclusions

In this paper, a nominal yet simple implicit spatial regularization (ISR) method was provided for face recognition via retaining as much spatial information between image pixels as possible. As opposed to existing explicit spatial regularization (ESR) for vector representation, our proposed ISR is based on a second-order tensor representation and retains spatial information through reshaping face image rather than constraining the projection vectors to be spatially smooth by introducing explicit regularization term, so ISR does not need to select regularization parameters. Compared with 2D methods and ESR, on the one hand, from the viewpoint of 2D methods, ISR takes advantages of the spatial information of the image, so more spatial information can be gained. On the other hand, from the perspective of ESR, ISR can avoid the awkward selection of regularization factor involved in the optimization objective and reduce the computational cost by inheriting from the 2D methods. As a result, ISR possesses the advantages of both ESR and 2D methods. Comprehensive experimental results demonstrated that the proposed ISR method is considerably competitive in face recognition accuracy to the explicit regularization method SSSL but with much lower computational cost.

Acknowledgment

This work was supported by the Natural Science Foundation of Jiangsu Province under Grant no. BK20130813, the National Natural Science Foundation of China under Grant nos. 61035003 and 61170151, the Fundamental Research Funds for the Central Universities under Grant no. NS2014100 and Jiangsu Qinglan project.

References

[1] K. Fukunage, Introduction to statistical pattern recognition. 2 ed., 1990.
[2] M. Turk, A. Pentand, Eigenfaces for recognition, J. Cogn. Neurosci. 3 (1) (1991) 71–86.
[3] A. Mashhoori, M. Zolghadri.Jahromi, Block-wise two-directional 2DPCA with ensemble learning for face recognition, Neurocomputing 108 (2) (2013) 111–117.
[4] J. Gui, et al., Discriminant sparse neighborhood preserving embedding for face recognition, Pattern Recognit. 48 (8) (2012) 2884–2893.

[5] X. Chen, et al., Spatially correlated nonnegative matrix factorization, NeuroComputing 139 (2) (2014) 15–21.
[6] Y. Pang, A.B.J. Teoh, F.S. Hiew, Locality regularization embedding for face verification, Pattern Recognit. 48 (1) (2015) 86–102.
[7] X. He, P. Niyogi. Locality preseving projections, in: Advances in Neural Information Processing Systems, 2003.
[8] D. Xu, et al., Marginal Fisher analysis and its variants for human gait recognition and content- based image retrieval, IEEE Trans. Image Process. 16 (11) (2007) 2811–2821.
[9] X. He, et al. Neighborhood preserving embedding, in: ICCV, 2005.
[10] H.T. Chen, H.W. Chang, T.L. Liu. Local discriminant embedding and its variants. Computer Vision and Pattern Recognition, 2005.
[11] Y. Gao, et al., 3-D object retrieval with Hausdorff distance learning, IEEE Trans. Ind. Electron. 61 (4) (2014) 2088–2098.
[12] Chen, K., J. Hennebert. Content-based image retrieval with LIRe and SURF on a smartphone-based product image database, in: (editor) José Francisco Martínez-Trinidad, Pattern Recognition, Springer, 2014, p. 231–240.
[13] Y. Gao, et al., 3-d object retrieval and recognition with hypergraph analysis, IEEE Trans. Image Process. 21 (9) (2012) 4290–4303.
[14] D. Cai, et al. Learning a spatially smooth subspace for face recognition, in: Computer Vision and Pattern Recogntion, 2007.
[15] J. Yang, et al., Two-dimensional PCA: a new approach to appearance-based face representation and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (1) (2004) 131–137.
[16] J. Ye. Generalized low rank approximations of matrices, in: Proceedings of the ICML Conference. 2004.
[17] M. Li, B.Z. Yuan, 2D LDA: a statistical lineardiscriminantanalysis for image matrix, Pattern Recognit. Lett. 26 (5) (2005) 527–532.
[18] J. Ye, R. Janardan, Q. Li. Two-dimensional linear discriminant analysis, in: NIPS, 2004.
[19] X. He, D. Cai, P. Niyogi. Tensor subspace analysis, in: Advances in Neural Information Processing Systems, 2005.
[20] W. Zhang, Z.C. Lin, X.O. Tang, Tensor linear Laplacian discrimination (TLLD) for feature extraction, Pattern Recognit. 42 (2009) 1941–1948.
[21] D. Cai, X. He, J.W. Han, Subspace Learning Based on Tensor Analysis, University of Illionois at Urbana Champaign Urbana (Report No: UIUCDCS-R-2005-2572), 2005.
[22] C.P. Hou, et al., Learning an orthogonal and smooth subspace for image classification, IEEE Signal Process. Lett. 16 (4) (2009) 303–306.
[23] W.M. Zuo, et al., Spatially smooth subspace face recognition using LOG and DOG penalties, Adv. Neural Netw. 5553 (2009) 439–448.
[24] S.C. chen, et al., Feature extraction approaches based on matrix pattern: MatPCA and MatFLDA, Pattern Recognit. Lett. 26 (2005) 1157–1167.
[25] S. Yan, et al. Element rearrangement for tensor-based subspace learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007.
[26] H.T. Chen, T.L. Liu, C.S. Fuh. Learning effective image metrices from few pairwise examples, in: Proceedings of the Internation Conference on Computer Vision, 2005.
[27] R. Kumar, et al., Trainable convolution filters and their application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1423–1436.
[28] R. Kumar, et al., Trainable convolution filters and their application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1423–1436.
[29] D. Hu, G. Feng, Z. Zhou, Two-dimensional locality preserving projections (2DLPP) with its application to palmprint recognition, Pattern Recognit. 40 (1) (2007) 339–342.
[30] K. Lee, J. Ho, D.J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005) 1–15.
[31] Q. Hong, S.C. Chen, X.L. Ni, Sub-pattern canonical correlation analysis with application in face recognition, Acta Autom. Sin. 34 (1) (2008) 21–30.

Yulian Zhu received her B.S and M.S degrees in computer application from Nanjing University of Aeronautics & Astronautics (NUAA) in 2001 and 2004, respectively. Then she worked in NUAA from April 2004. There she received a Ph.D. degree in computer application in 2010. Her main research interests include machine learning, pattern recognition, and image processing.

Songcan Chen received the B.S. degree from Hangzhou University (now merged into Zhejiang University), the M.S. degree from Shanghai Jiaotong University and the Ph.D. degree from Nanjing University of Aeronautics and Astronautics (NUAA) in 1983, 1985, and 1997, respectively. He joined in NUAA in 1986, and since 1998, he has been a full-time Professor with the Department of Computer Science and Engineering. He has authored/co-authored over 170 scientific peer-reviewed papers and ever obtained Honorable Mentions of 2006, 2007 and 2010 Best Paper Awards of Pattern Recognition Journal respectively. His current research interests include pattern recognition, machine learning, and neural computing.

Qing Tian received the B.S. degree in computer science from Southwest University for Nationalities, China, and the M.S. degree in computer science from Zhejiang University of Technology, China, respectively with the honors of Sichuan province-level excellent graduate and Zhejiang province-level excellent graduate in 2008 and 2011. From Feb 2011 to Feb 2012, as a researcher in the field of gender/age recognition, he worked in Arcsoft, U.S. Now he is a Ph.D. candidate in computer science at Nanjing University of Aeronautics and Astronautics, and his current research interests include machine learning and pattern recognition.