Handwriting Representation and Recognition through a Sparse Projection and Low-Rank Recovery Framework

Zhao Zhang [a, b, c, *], Cheng-Lin Liu [b], and Mingbo Zhao [c]
[a] School of Computer Science and Technology, Soochow University, Suzhou 215006, P. R. China
[b] National Laboratory of Pattern Recognition, Chinese Academy of Sciences, Beijing 100190, P.R. China
[c] Department of Electronic Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong
*Correspondence author: E-mail: cszzhang@gmail.com

Abstract— **This paper proposes a Robust Principal Component Analysis (RPCA) based framework called Sparse Projection and Low-Rank Recovery (SPLRR) for representing and recognizing handwritings. SPLRR calculates a similarity preserving sparse projection for salient feature extraction and processing new data for classification in addition to delivering a low-rank principal component and identifying errors or missing pixel values from a given data matrix. As a result, SPLRR will be applicable for handwritten recovery, recognition and the applications requiring online computation. To encode the similarity between features in the learning process, the Cosine similarity based regularization is incorporated to the SPLRR formulation. The sparse projection and the lowest-rank components are calculated from a scalable convex minimization problem that can be efficiently solved in polynomial time. The effectiveness of the proposed SPLRR is examined by handwritten digital repairing, stroke correction and recognition on two real problems. Results show that SPLRR can deliver state-of-the-art results in handwriting representation.**

Keywords- Handwriting representation; Low-rank recovery; Sparse projection; Handwriting recognition

I. INTRODUCTION

In various practice applications, such as face and handwriting recognition, numerous high-dimensional data (e.g., documents) that can be characterized using low-rank subspaces are often handled. In many cases, these real data contains corruptions or missing pixel values, which arouses considerable attention on the issue of automatically recovering the low-rank structures and fitting errors, such as [1-12]. One most representative low-rank recovery method is called *Robust Principal Component Analysis* (RPCA) [3], [4], [8] which is widely used in various areas, including face recovery [7], image tag refinement [36], document analysis [37], and texture transformation [15]. More specifically, for a given matrix $X=[x_1,x_2,...,x_N]\in\mathbb{R}^{n\times N}$ corrupted by some errors E_0, RPCA exactly recovers X_0 ($X=X_0+E_0$) by solving the following convex problem:

$$\left(Y^*,E^*\right)=\arg\min_{Y,E}\left(\|Y\|_*+\lambda\|E\|_\ell\right),\ \text{Subj}\ X=Y+E\ , \qquad (1)$$

where $\|\bullet\|_*$ is the nuclear norm [8], $\|\bullet\|_\ell$ denotes the $l^{2,1}$-norm, l^1-norm or the squared Frobenius norm for characterizing the errors E, and λ is a parameter. From Eq.1, the optimal Y^*, called the principal components of *X,* is the low-rank recovery to X_0. RPCA can address the gross corruptions well with large magnitude if only a fraction of entries are corrupted [8], but it is essentially a transductive algorithm, i.e., it cannot represent new data and requires recalculating over all data when facing new samples [12]. To overcome this issue, a generalization of RPCA, referred to as *Inductive Robust Principal Component Analysis* (IRPCA) [12], was recently proposed. Unlike RPCA, IRPCA targets on error correction through seeking a low-rank projection matrix $Q\in\mathbb{R}^{n\times n}$ to remove the possible corruptions from a given set of observations. For a given data matrix *X*, IRPCA learns the lowest-rank projection Q and the principal components $Y=[y_1,y_2,...,y_N]$ by solving the following convex nuclear norm problem:

$$\left(Q^*,E^*\right)=\arg\min_{Q,E}\left(\|Q\|_*+\lambda\|E\|_1\right),\ \text{Subj}\ X=Y+E, Y=QX\ , \qquad (2)$$

from which the optimal solution Q^* can be obtained, then the sparse errors can be estimated as $X-Q^*X$. Based on the learnt projection matrix Q^* from training data, IRPCA can represent new input data and effectively remove the possible corruptions through projecting the datum onto the subspace spanned by Q^* [12], but note that the similarities among compact features to be encoded are lost in the IRPCA formulation.

We in this paper propose a new unified sparse projection and nuclear norm minimization framework, referred to as *Sparse Projection and Low-Rank Recovery* (SPLRR), for recovering low-rank matrices, extracting saliency features and correcting errors at the same time. Our proposed SPLRR framework is a generalization to RPCA via including a similarity preserving sparse feature extraction term, which is mainly proposed for processing handwriting data, which is motivated by two facts: 1) The stroke directions and shapes of the handwritten digits or characters are usually irregular and even curved because the writing styles of different writers varied a lot [26]. As a result, the handwritings can bear some corruptions or missing stroke regions to be repaired; 2) The irregularity of the stroke shapes precisely reflect the writing styles and hence may be useful in identifying personal handwritings.

Given a set of handwriting vectors *X* with some corruptions or missing values, the objective of this paper is to decompose it to a low-rank component encoding the principal features to be recovered, a sparse component encoding the visually salient stroke features that are able to best distinguish handwritings in classification task, and an error part. Formally, for handwriting

data containing certain errors, this paper mainly considers the following handwritten recovery problem:

Problem 1 (Handwritten recovery and salient stroke feature extraction): For handwritten data x expressed as

$$x = r + f + e, \text{ where } x, r, f, e \in \mathbb{R}^n, \quad (3)$$

the goal is to automatically identify and recover the low-rank principal component r by regenerating missing values, extract visually salient stroke features f for classifying handwriting, and correct the possible errors e. An important aspect of our SPLRR is to seek a similarity preserving sparse projection to extract the saliency features from training samples, so *Sparse Representation* (SR) [17], [19] is a related criterion. This work considers the following convex problem to seek the sparsest representation R of all data vectors jointly:

$$\left(R^*, E^*\right) = \arg\min_{R,E} \left(\|R\|_1 + \lambda\|E\|_\ell\right), \text{Subj } X = DR + E, Diag(R) = 0, \quad (4)$$

where $Diag(R) = 0$ is to avoid the trivial solution $R = I$, and the matrix X itself is usually set to be the dictionary D. SR has attracted many interests and promising results are delivered in the context of image representation and classification, e.g. [16-20]. Most importantly, the sparsest representation can exhibit a natural discriminating power [17], [19]. But estimating an optimal D for SR is still difficult and SR is also a transductive criterion, i.e., it cannot represent new vectors for classification.

We highlight the three contributions of this present paper as follows. First, we technically propose an effective framework termed SPLRR for handwriting recovery and recognition via fully considering the properties of handwritings. The SPLRR problem can be solved in polynomial time. Second, for salient stroke feature extraction, we present a variant of regular SR to calculate a sparse projection for embedding new handwriting data in recognition. To preserve the similarity among features, a Cosine similarity based regularized term is incorporated in this setting. Third, we mainly evaluate SPLRR for handwritten recovery, stroke correction and recognition.Satisfactory results for handwriting representation are delivered by our SPLRR.

The paper is summarized as follows. Section II reviews the related studies. In Section III, we present the SPLRR method and its implementation. The connections between SPLRR and the related criteria are also detailed. We in Section IV conduct simulations to examine SPLRR for handwriting representation and recognition. Finally, Section V concludes the paper.

II. Brief Reviews Of Related Works

Nuclear-norm minimization based recovery criteria are now standard, such as [1-12], of which RPCA is one most popular one and has been widely used in many areas. To better process mixed data, *Low-Rank Representation* (LRR) [6] was recently proposed. LRR aims at seeking the lowest-rank representation $Z = [z_1, z_2, ..., z_N] \in \mathbb{R}^{N \times N}$ among all the candidates that represent all data vectors as the linear combination of the bases in a given dictionary D. By setting the matrix X itself as the dictionary D, the problem of LRR is formulated as

$$\left(Z^*, E^*\right) = \arg\min_{Z,E} \left(\|Z\|_* + \lambda\|E\|_\ell\right), \text{ Subj } X = XZ + E. \quad (5)$$

Clearly, the LRR criterion reduces to RPCA if $D = I$. With the solution $\left(Z^*, E^*\right)$ calculated, the original data are recovered by $X - E^*$ (or XZ^*). Unlike RPCA, LRR focuses on solving the subspace clustering and segmentation problem by Z^*. Notice that LRR shares a similar form as the SR problem, that is SR can perform data recovery and subspace clustering similarly as LRR does [6][35]. More recently, *Zhou et al.* have studied the random projections based approximated low-rank and sparse decomposition of a matrix and developed a new method called *Go Decomposition* (GoDec) [5], formulated as

$$\left(Y^*, A^*\right) = \arg\min_{Y,A} \left(\|X - A - S\|_F^2\right), \text{Subj } rank(Y) \le r, card(A) \le \alpha, \quad (6)$$

where r and α are constants, and $card(A)$ is the number of nonzero entries in matrix A. Note that a common shortcoming of RPCA, SR, LRR and GoDec is that they are essentially transductive methods, i.e. they have to recalculate over all data when new samples are input. This causes heavy computational burden and hinders fast computation in the online applications. Moreover, when the available data is corrupted by dense noise, the performance of SR and LRR is largely influenced by the choice of dictionary [6]. Although some efforts on dictionary learning have been made (e.g., [18], [33]), the optimization of an informative dictionary is still difficult.

IRPCA was recently proposed to generalize RPCA method by calculating a low-rank projection. Unlike RPCA, IRPCA targets on correcting the possible corruptions [12]. Although IRPCA can well address the random corruptions, it is unable to characterize the outliers and sample-specific corruptions to be addressed in this work. Note that IRPCA is a special case of LRR [12]. *Latent LRR* (LatLRR) [31], as a combination of LRR and IRPCA, is another related algorithm which seeks to construct the dictionary using the observed data X and hidden unobserved data X_H, i.e., $D = [X, X_H]$, and recovers the hidden effects by solving the following convex problem:

$$\left(Z^*, E^*\right) = \arg\min_{Z,E} \left(\|Z\|_* + \lambda\|E\|_\ell\right), \text{Subj } X = DZ + E, D = [X, X_H]. \quad (7)$$

LatLRR can resolve the issue of insufficient sampling and improve the robustness to noise [6], [31] via recovering the hidden effects to some extent. Although LatLRR and IRPCA are cable of computing a projection matrix that can be applied for feature extraction and embedding new coming data vectors in classification, they are different from the formulation of our SPLRR in two aspects. First, the projection matrix of LatLRR and IRPCA are essentially low-ranked, while the one obtained by SPLRR is sparse. Second, the similarities among compact features to be encoded are lost in LatLRR and IRPCA as well as the other aforementioned criteria. In contrast, our SPLRR clearly takes the similarity preservation capability of features into account in addition to calculate a sparse projection matrix for saliency feature extraction.

III. Problem Formulation

A. Sparse Projection and Low-Rank Recovery Framework

For a given set of handwriting vectors $X = [x_1, x_2, ..., x_N] \in \mathbb{R}^{n \times N}$ corrupted by errors, SPLRR calculates a similarity preserving

sparse projection $S \in \mathbb{R}^{n\times n}$, and decomposes X into a low-rank component $L \in \mathbb{R}^{n\times N}$, a sparse component SX encoding salient stroke features and an error part E by solving

$$\begin{aligned}&\left(L^*, S^*, E^*\right)=\\&\arg\min_{L,S,E}\left((1-\beta)rank(L)+\beta\|S\|_0+\xi\widehat{f}_S(X)+\lambda\|X-Y\|_{2,0}\right),\\&\text{Subj } Y=L+SX\end{aligned} \quad (8)$$

where $\|\bullet\|_{2,0}$ is the $l^{2,0}$-norm, $\|\bullet\|_0$ is the l^0-norm, $\beta\in[0,1]$ is a parameter for trading-off the low-rankness and the sparsity, $\lambda>0$ is a parameter relying on the true levels of errors, $X-Y$ identifies errors E, and $\widehat{f}_S(X)$ with a nonnegative parameter ξ denotes a regularized term that is integrated to encode the similarity between the saliency features. The above problem is not directly tractable because of the discrete nature of the rank function, l^0-norm and $l^{2,0}$-norm. As a common practice, we replace the rank function with nuclear norm. Also, by relaxing $l^{2,0}$-norm and l^0-norm with the $l^{2,1}$-norm $\|\bullet\|_{2,1}$ and l^1-norm $\|\bullet\|_1$ respectively, we have the following convex surrogate for the above optimization problem:

$$\begin{aligned}\left(L^*, S^*, E^*\right)=\arg\min_{L,S,E}\left((1-\beta)\|L\|_*+\beta\|S\|_1+\xi\widehat{f}_S(X)+\lambda\|E\|_{2,1}\right),\\ \text{Subj } X=L+SX+E\end{aligned} \quad (9)$$

where the matrix nuclear norm $\|\bullet\|_*$, matrix l^1-norm $\|\bullet\|_1$ and $l^{2,1}$-norm $\|\bullet\|_{2,1}$ are respectively defined as

$$\|L\|_* := \sum_i \sigma_i(L),\ \|S\|_1=\sum_{i,j}\left|S_{i,j}\right|,\ \|E\|_{2,1}=\sum_{j=1}^{N}\sqrt{\sum_{i=1}^{N}\left([E]_{i,j}\right)^2}, \quad (10)$$

where $\sum_i \sigma_i(L)$ is the sum of the singular values of matrix L. For handwritten salient stroke features SX, we define $\widehat{f}_S(X)$ as follows to preserve the similarity between them:

$$\begin{aligned}\widehat{f}_S(X)&=\frac{1}{2}\sum_{i,j=1}^{N}W_{i,j}^{(t)}d^2\left(Sx_i,Sx_j\right)\\&=\frac{1}{2}\sum_{i,j=1}^{N}W_{i,j}^{(t)}Tr\left(\left(Sx_i-Sx_j\right)\left(Sx_i-Sx_j\right)^{\mathrm{T}}\right)=Tr\left(SG^{(t)}S^{\mathrm{T}}\right)\end{aligned}, \quad (11)$$

where $d^2\left(\widetilde{x}_i,\widetilde{x}_j\right)=\left\|Sx_i-Sx_j\right\|^2$ is the squared Euclidean distance between the salient handwriting features Sx_i and Sx_j, $G^{(t)}=X\left(Q^{(t)}-W^{(t)}\right)X^{\mathrm{T}}=XF^{(t)}X^{\mathrm{T}}$ is a symmetric matrix, $\|\bullet\|$ denotes the l^2-norm, and $Q_{ii}^{(t)}=\sum_j W_{i,j}^{(t)}$. In this paper, we apply the Cosine similarity to define the edge weight matrix $W^{(t)}$ for encoding the similarity between handwritten features, where the similarity is measured by the angle between them. Note that the Cosine similarity was widely applied for comparing documents in text mining [27], [28]. For given handwritten vectors x_i and x_j, the matrix $W^{(t)}$ is weighted as

$$W_{i,j}^{(t)}=\exp\left(\cos(\vartheta)\right),\ where\ \cos(\vartheta)=\left\langle x_i,x_j\right\rangle/\left(\|x_i\|\|x_j\|\right), \quad (12)$$

where the inner product $\left\langle x_i,x_j\right\rangle=x_i^{\mathrm{T}}x_j$, and the resulting similarity values are within $\left[e^0,e^1\right]$, i.e., the higher of the values, the closer of the corresponding handwriting pairs. Note that the regularized technique is widely applied in machine learning, e.g., the Laplacian regularization in the sparse coding for encoding the locality between features [18], [20]. But k-nearest neighbor search [25][34] is involved in these studies, but estimating an optimal k for locality preservation is difficult. From Eq.9, we refer to the minimizer L^* as the lowest-rank recovery to the original data,and the optimal S^* as the sparsest representation of X. Moreover, for given a new vector Δ, the saliency features is extracted by $S^*\Delta$ and the errors can be estimated as $\Delta-L^*-S^*\Delta$. That is, the sparse projection S^* can be used for feature extraction in classification.

B. Efficient Solution by Convex Optimization

The proposed SPLRR formulation in Eq.9 is convex, thus it can be solved by using various methods, e.g., the Argument Lagrange Multiplier (ALM) or the inexact ALM [9]. We first convert Eq.9 to the following equivalent one:

$$\begin{aligned}&\left(L^*, W^*, S^*, E^*\right)=\\&\arg\min_{L,W,S,E}\left((1-\beta)\|L\|_*+\beta\|W\|_1+\xi Tr\left(SG^{(t)}S^{\mathrm{T}}\right)+\lambda\|E\|_{2,1}\right),\\&\text{Subj } S=W,\ X=L+SX+E\end{aligned} \quad (13)$$

where $Tr(\bullet)$ is trace operator. The corresponding augmented Lagrangian function $\wp$ is addressed as

$$\begin{aligned}&\wp(L,W,S,E,Y_1,Y_2,\mu)=\\&(1-\beta)\|L\|_*+\beta\|W\|_1+\xi Tr\left(SG^{(t)}S^{\mathrm{T}}\right)+\lambda\|E\|_{2,1}+\left\langle Y_1,S-W\right\rangle,\\&+\left\langle Y_2,X-L-SX-E\right\rangle+\frac{\mu}{2}\left(\|S-W\|_F^2+\|X-L-SX-E\|_F^2\right)\end{aligned} \quad (14)$$

where Y_1, Y_2 are Lagrange multipliers, μ is a positive scalar, and $\|\bullet\|_F$ is the matrix Frobenius norm. Notation A^{T} denotes the transpose of the matrix A. The ALM algorithm alternately updates variables L, W, S and E, by iteratively minimizing the augmented Lagrangian function $\wp$:

$$\begin{aligned}&\left(L_{k+1},W_{k+1},S_{k+1},E_{k+1}\right)=\arg\min_{L,W,S,E}\wp_k\left(L,W,S,E,\mu\right)\\&Y_1^{k+1}=Y_1^k+\mu_k\left(S_{k+1}-W_{k+1}\right)\\&Y_2^{k+1}=Y_2^k+\mu_k\left(X-L_{k+1}-S_{k+1}X-E_{k+1}\right)\end{aligned} \quad .(15)$$

As elaborated in [21], [22], the iteration converges to the optimal solution of the optimization in Eq.13 when $\{\mu_k\}$ is a monotonically increasing. Since variables L, W, S and E are dependent on each other, the above problem cannot be solved directly. In this work, we update the variables alternately with others fixed by iteratively solving the sub-problems:

$$\begin{aligned}&L_{k+1}=\arg\min_L\wp_k\left(L,W_k,S_k,E_k,Y_1^k,Y_2^k\right)\\&W_{k+1}=\arg\min_W\wp_k\left(L_{k+1},W,S_k,E_k,Y_1^k,Y_2^k\right)\\&S_{k+1}=\arg\min_S\wp_k\left(L_{k+1},W_{k+1},S,E_k,Y_1^k,Y_2^k\right)\\&E_{k+1}=\arg\min_E\wp_k\left(L_{k+1},W_{k+1},S_{k+1},E,Y_1^k,Y_2^k\right)\end{aligned}, \quad (16)$$

where each step corresponds to a convex problem that can be effectively solved. The convergence properties of our SPLRR algorithm are similar in sprit as those of [6], [12], [22]. Note that the above sub-problems all have closed-form solutions. L and W are respectively solved by applying the Singular Value Thresholding (SVT) operator [10] and the shrinkage operator [9]. According to [6], [8], the i-th column $E_{k+1}^{:,i}$ of solution E_{k+1} at the (k+1)-iteration can be calculated as

$$E_{k+1}^{:,i}=\begin{cases}\left[\left(\|\widehat{\Phi}_{:,i}^{E}\|_2-(\lambda/\mu_k)\right)/\|\widehat{\Phi}_{:,i}^{E}\|_2\right]\widehat{\Phi}_{:,i}^{E} & if\ (\lambda/\mu_k)<\left\|\widehat{\Phi}_{:,i}^{E}\right\|_2\\ 0 & otherwise\end{cases}, \quad (17)$$

where $\widehat{\Phi}^{E}=X-L_{k+1}-S_{k+1}X+Y_2^k/\mu_k$, and $\widehat{\Phi}_{:,i}^{E}$ is the *i*-th column of matrix $\widehat{\Phi}^{E}$. Note that SPLRR can be similarly implemented as RPCA and IRPCA.The major computation of SPLRR is at solving *L*, which requires calculating the SVD of matrices, so the complexity of our SPLRR is the same as that of the inexact ALM based RPCA method. In this work, we set $\eta=1.12$ for SPLRR. Note that the convergence analysis of the ALM has been well established in [9], but when there are more than two terms in the minimizations, for instance [6], [12], [22] and our proposed problem, a rigorous proof of the convergence still remains a difficult issue [6], [22]. In our simulations, we can observe that our SPLRR always converges to the problem with the iteration number ranged from 30 to 280.

C. Discussion: Connection and Comparison

This subsection discusses some issues relating to SPLRR.

(1) Connection with RPCA. Our SPLRR is a generalization of RPCA. Clearly, by setting $\beta=\xi=0$, the SPLRR formulation is reduced to RPCA when the same norm is imposed on the term *E* for modeling the errors.

(2) Connection with GoDec [5]. Our SPLRR formulation is a generalization of GoDec criterion. Since $X-L-SX$ identifies the errors in SPLRR, by imposing the squared Frobenius norm on term *E* to model noise [6] and using similar formulation method as GoDec, the objective function of our SPLRR can be reformulated as the following framework:

$$\left(L^*,S^*,E^*\right)=\arg\min_{L,S,E}\left(\|X-L-SB\|_F^2+\xi\widehat{f}_S(X)\right)\ \text{Subj}\ rank(L)\le r,\ card(S)\le\alpha. \quad (18)$$

Clearly, if $B=X$, the above formulation corresponds to our SPLRR. By setting $B=I$ (i.e. the standard basis) and $\xi=0$ the above problem identifies GoDec, thus GoDec is regarded as a special case of our SPLRR formulation.

(3) Connections with the SR [17], ***Graph Regularized Sparse Coding*** **(GraphSC)** [18] **and** ***Laplacian Sparse Coding*** **(LSc)** [20]. In this case, we set $\beta=1$, i.e., SPLRR focuses on sparse coding. By imposing the squared Frobenius norm on the error term *E*, SPLRR is equivalent to the following problem:

$$S^*=\arg\min_{S}\left(\|X-SX\|_F^2+\widehat{\varpi}_1\sum_i\|s_i\|_1+\widehat{\varpi}_2\widehat{f}_S(X)\right), \quad (19)$$

where $\widehat{\varpi}_1=1/\lambda$, $\widehat{\varpi}_2=\xi/\lambda$ denote two control parameters, and $\widehat{f}_S(X)=(1/2)\sum_{i,j}W_{i,j}^{(t)}\|Sx_i-Sx_j\|^2=Tr\left(SXF^{(t)}X^{\mathrm{T}}S^{\mathrm{T}}\right)$.Motivated by [12], the above optimization problem can be reformulated as the following equivalent one:

$$S^*=\arg\min_{S}\left(\sum_i\left\|x_i^{\mathrm{T}}-X^{\mathrm{T}}s_i^{\mathrm{T}}\right\|_F^2+\widehat{\varpi}_1\sum_i\left\|s_i^{\mathrm{T}}\right\|_1+\widehat{\varpi}_2 f_S(X)\right), \quad (20)$$

where $f_S(X)$ is the transpose of $\widehat{f}_S(X)$. Note that setting X^{T} itself as the dictionary $D=[D_1,D_2,...,D_N]$ may not be optimal in the sparse coding formulations [18], [20], thus one usually alternately learns the dictionary and the sparse codes. Next we reformulate the above problem as

$$\left(D^*,S^*\right)=\arg\min_{D,S}\left(\sum_i\left\|x_i^{\mathrm{T}}-Ds_i^{\mathrm{T}}\right\|_F^2+\widehat{\varpi}_1\sum_i\left\|s_i^{\mathrm{T}}\right\|_1+\widehat{\varpi}_2 f_S(X)\right)\ \text{Subj}\ \|D_i\|^2\le c,\ i=1,2,...,N, \quad (21)$$

where *c* denotes a constant, e.g., constant 1. In this case, we can equivalently minimize $f_S(X)=(1/2)\sum_{i,j}W_{i,j}^{(t)}\|Ds_i^{\mathrm{T}}-Ds_j^{\mathrm{T}}\|^2=(1/2)\sum_{i,j}W_{i,j}^{(t)}Tr\left(\left(Ds_i^{\mathrm{T}}-Ds_j^{\mathrm{T}}\right)^{\mathrm{T}}\left(Ds_i^{\mathrm{T}}-Ds_j^{\mathrm{T}}\right)\right)=Tr\left(SD^{\mathrm{T}}F^{(t)}DS^{\mathrm{T}}\right)$ to preserve the similarity of features in the sparse coding process with *D* fixed. Also, by applying the *k*-neighborhood to find the neighbors of each sample and constructing the weight matrix $W^{(t)}$ by the simple-minded method or the heat kernel [34], i.e., $W_{i,j}^{(t)}$ are weighted if x_i belongs to the *k*-neighbors of sample x_j, the problem in Eq.21 further becomes

$$\left(D^*,S^*\right)=\arg\min_{D,S}\left(\left\|X^{\mathrm{T}}-DS^{\mathrm{T}}\right\|_F^2+\widehat{\varpi}_1\sum_i\left\|s_i^{\mathrm{T}}\right\|_1+\widehat{\varpi}_2\sum_{i,j}W_{i,j}^{(t)}\left\|Ds_i^{\mathrm{T}}-Ds_j^{\mathrm{T}}\right\|^2\right)\ \text{Subj}\ \|D_i\|^2\le c,\ i=1,2,...,N. \quad (22)$$

Clearly, the above optimization is equivalent to the GraphSC criterion when *D* is known, especially when $D=I$. By further imposing a constraint $\|D_i\|^2=1$ to avoid the scaling problem of D_i, the above problem is also identified to the formulation of LSc. Thus, the optimal solution S^* to the above problem can be similarly calculated as GraphSC and LSc. Note that if the regularization term $f_S(X)$ and weight matrix $W^{(t)}$ are defined as above, and each row of the data matrix *X* corresponds to a data point, by setting dictionary $D=X^{\mathrm{T}}$, the above formulation can be converted to the optimization problem of Eq.19. That is, our SPLRR with $\beta=1$ can be considered as a special case of the formulations of GraphSC and LSc. But note that, unlike GraphSC and LSc, SPLRR avoids estimating an optimal *k* and heat kernel width for similarity preservation, since this issue is very difficult in reality. By setting $\widehat{\varpi}_2=0$ or $\xi=0$, the above formulation is further transformed to

$$\left(D^*,S^*\right)=\arg\min_{D,S}\left(\left\|X^{\mathrm{T}}-DS^{\mathrm{T}}\right\|_F^2+\widehat{\varpi}_1\sum_i\left\|s_i^{\mathrm{T}}\right\|_1\right)\ \text{Subj}\ \|D_i\|^2\le c,\ i=1,2,...,N. \quad (23)$$

Since $X^{\mathrm{T}}-DS^{\mathrm{T}}$ identifies the errors, if we similarly consider each row of the data matrix *X* as an instance, and set $D=X^{\mathrm{T}}$ and $\widehat{\varpi}_1=\lambda=1$, the above sparse coding problem is identified to the SR formulation in Eq.4. In other words, the formulation of our proposed SPLRR method with $\beta=1$ and $\widehat{\varpi}_2=0$ can also be treated as a special example of the SR formulation in Eq.4 when imposing the same constraint.

IV. Simulation Results and Analysis

A. Baselines and Settings

In this section, we mainly evaluate SPLRR for handwriting recovery and recognition. For recognition, the performance is compared with *Projective Nonnegative Matrix Factorization* (PNMF) [24], IRPCA, and LatLRR.

(a) *PNMF*: Euclidean PNMF approximates a given set of vectors $\{x_i\}_{i=1}^N$ by its nonnegative subspace projection, and factorizes a projection matrix P with given rank *r* to a non-

negative low-rank $H \in \mathbb{R}^{n \times r}$ and its transpose by minimizing the following reconstruction error [24]:

$$H^* = \arg\min_H (1/2)\|X - HJ\|_F^2, \text{Subj } J = H^{\mathrm{T}} X, H > 0 . \quad (24)$$

After calculating H, a nonnegative projection $P = HH^{\mathrm{T}}$ can be constructed for projecting new data and PX can be treated as the reconstruction of the original data.

(b) *Data Preparation*. Two handwriting digits databases, i.e., MNIST [14] and CASIA-HWDB1.1 [29], are tested. As the sizes of images in CASIA-HWDB1.1 are inconsistent, we resize all images to 14×14 pixels. A subset called HWDB1.1-D, including 2381 digits ('0'-'9'), from CASIA -HWDB1.1 is sampled. MNIST has a training set of 60000 samples and a test set of 10000 samples. In this study, we choose first 2000 samples from the training set and first 2000 samples from the test set as our experimental set, in which each digit has about 400 handwriting samples. We show typical samples in Figure 1 and describe the datasets in Table 1.

(c) *Parameter Settings*. For fair comparison, the $l^{2,1}$-norm is regularized on the term E of SR, IRPCA, LatLRR and SPLRR. Notice that these techniques have a common parameter λ to estimate. This paper mainly selects $\lambda \in \{4 \times 10^i \mid i = -4, -3, ..., -1\}$. For each algorithm, various parameters are tested and the best results over tuned parameters are reported. The number r in PNMF is set to the subject number as [24]. SPLRR has three parameters, i.e., β, ξ, λ. In the simulations, if X itself is an image, we set ξ to 0 and apply the idea of the "grid-search" method [30] on β and λ by repeating experiments. Various pairs of (β, λ) values are tested and the one according to the best performance is picked. Otherwise if X denotes a set of data vectors, we fix ξ and similarly perform grid-search on β and λ. This case sets $\xi \in \{10^i \mid i = -4, -3, ..., 2\}$ and picks the pairs of (β, λ) values according to the best test performance. We perform all simulations on PC with Intel (R) Core (TM) Duo CPU E8400 @ 3.00 GHz 2.99 GHz 1.96G.

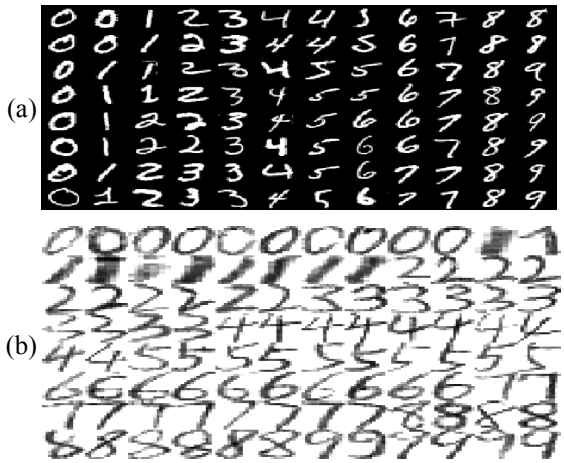


Figure 1: Typical images of (a) MNIST, (b) HWDB1.1-D.

Table 1: The specifications of the handwriting databases.

Dataset Name	*# Class*	*# Num.*	*# Dim.*
MNIST	C=10	70000	28×28
HWDB1.1-D	C=10	2381	14×14

(d) *Evaluation Metric*. We visually evaluate the handwriting recovery results. For handwriting recognition, Euclidian one-nearest-neighbor (1NN) classifier is applied for quantitatively evaluating the simulation results. For recognition, each dataset is randomly split into training and test sets. After calculating the projection matrix from training set, test data are projected onto the directions. Finally, the embedded data are identified by a NN classifier and the averaged result is shown. *Principal Component Analysis* (PCA) [23] is used to eliminate the null space of training set before feature extraction.

B. Document Recovery

This experiment aims to visualize the performance of SPLRR for document representation, including recovery and feature extraction. Two simulations over different document data are

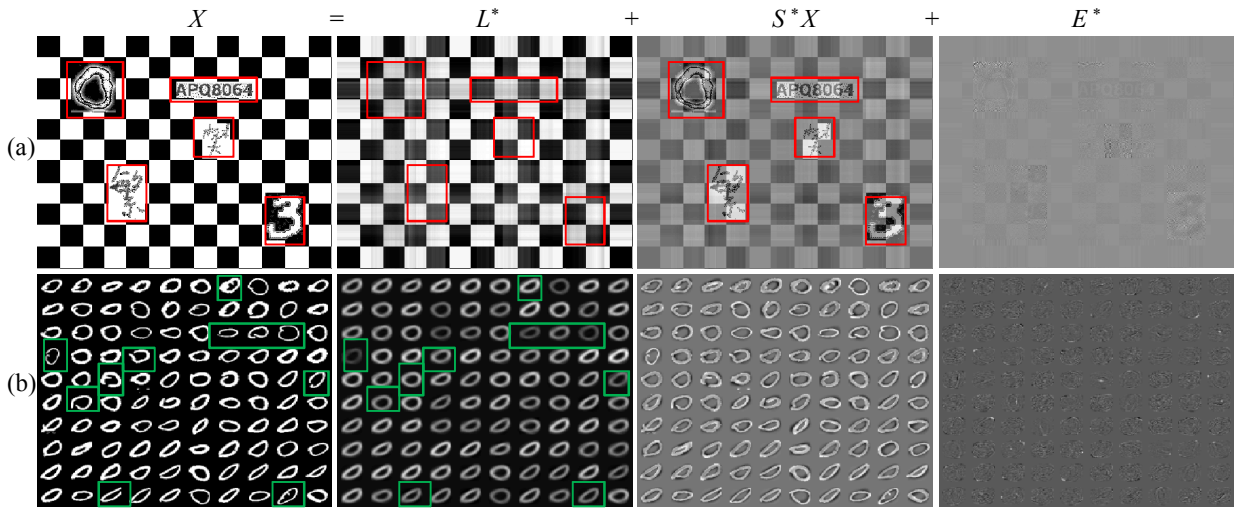


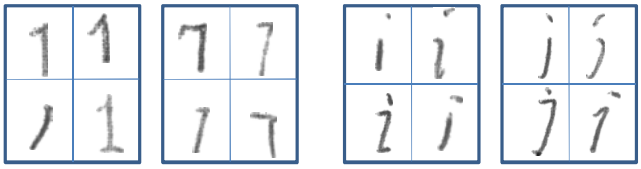
Figure 2: Recovery of SPLRR on: (a) Background corrupted by documents, (b) MNIST digit '0'.

conducted. Given a matrix X, SPLRR handle it by recovering a low-rank principal component L^*, extracting salient stroke features S^*X, and delivering a sparse part E^* fitting errors. We first test SPLRR for recovering the background from the heavy pixel corruptions, including handwritten digits, Chinese and English characters at the foreground. The corruptions in the matrix X are denoted by red rectangles, and the recovery results are illustrated in Figure 2(a). Seeing from the results, we see that SPLRR effectively removes the pixel corruptions and repairs the background. But conversely, the corruptions in the background may be regarded as the salient discriminative information for distinguishing backgrounds in classification problem. Note that our SPLRR can provide a solution to this problem. Specifically, based on the sparse projection matrix S^*, the visual salient strokes can be efficiently extracted by our SPLRR. Moreover, the sparse projection S^* can be used to represent new test data in classification.

We next address another simulation to examine SPLRR for processing handwriting digits with certain missing values or errors. This task is mainly to evaluate SPLRR for recovering the missing values and extracting the salient stroke features at the same time. In this study, the MNIST database is employed and we aim at visualizing 100 images of the handwritten digit "0". Note that matrix composed by transforming the images as column data vectors is naturally of low-ranked. The results are shown in Figure 2(b), where we use the rectangles to denote the digits with missing pixel values in the original matrix. We observe from the results that our proposed SPLRR method can successfully recover the missing pixel values and removes the noise, along with delivering salient features.

C. Handwritten Stroke Correction

In this simulation, we further evaluate our SPLRR for isolated handwriting (such as handwritten digits or characters) stroke correction by data recovery. It is worth noting that handwritten stroke correction is important and challenging in handwriting image analysis because of the variability of writing styles of different writers. That is, the stroke shapes of the handwritten digits or handwritten characters are usually irregular and un-uniformly curved [26]. Thus, those originally similar digits or characters, e.g., (digits '1' vs. '7') and (letters '*i*' vs. '*j*') from CASIA-HWDB1.1, are easier to be misclassified. We have shown some typical examples of this kind in Figure 3.



(a) '1' vs. '7' (b) '*i*' vs. '*j*'

Figure 3: Irregular, similar handwritten digits and characters.

This study is prepared to test SPLRR for correcting the incorrect or irregular strokes. In this study, we visualize the stroke correction results of 100 images of handwritten digit '7' from the MNIST and HWDB1.1-D databases respectively, in Figure 4. We have also shown examples of the irregular or incorrect strokes using the rectangular boxes in the first column of Figure 4. Observing from the results, we find the incorrect or irregular strokes of the handwritings from each set are accurately recovered by our SPLRR, as illustrated by L^*. But again one must note that these irregular strokes may also be useful in some specific applications, for instance personal handwriting identification [33], as it can precisely reflect the writing styles of a person, which is a factor that is as important as successfully repairing the irregular or incorrect strokes. In other words, effectively extracting the salient stroke features of the handwritings and performing stroke correction is equally important. As observed from the results in Figure 4, using our proposed SPLRR can enhance the visual saliency of writing strokes of different persons in addition to recover the irregular strokes and remove the sparse errors.

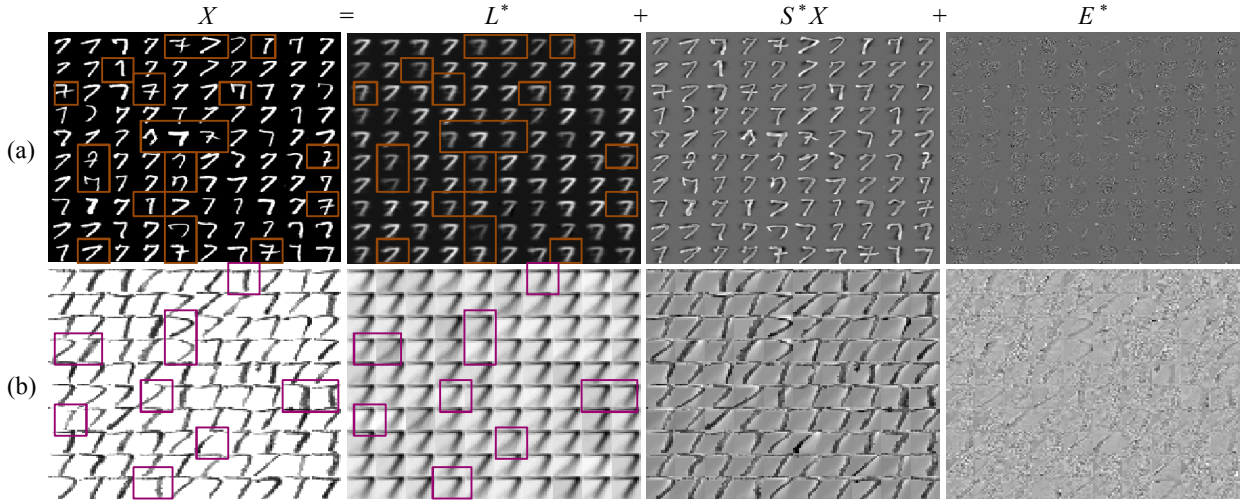


Figure 4: Handwritten stroke correction results of SPLRR on: (a) MNIST, (b) HWDB1.1-D.

D. Handwritten Digits Recognition

We then test SPLRR for handwritten digits recognition on the two databases. For each dataset, we normalize the handwritten pixel values of the images to [0, 1]. As a common practice, this study resizes all images of MNIST to 16×16 pixels due to the computational consideration. In this simulation, we mainly compare the performance of SPLRR with IRPCA, LatLRR and PNMF, that is, we directly use the projection matrix of SPLRR, IRPCA, LatLRR and PNMF learnt from the training data for embedding test data for classification by projecting the test data onto the projections.

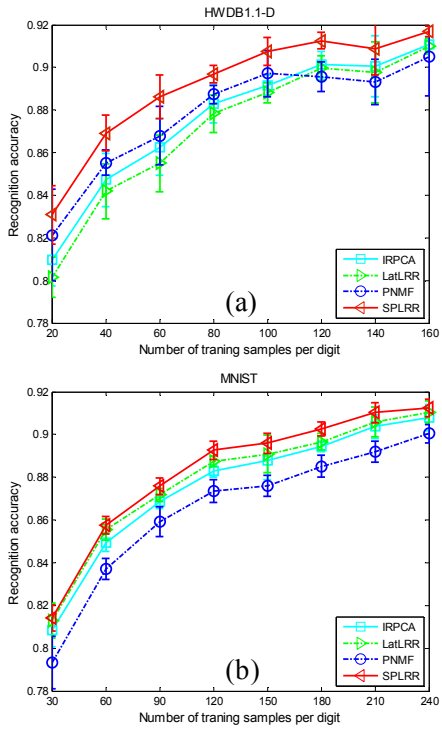


Figure 5: Performance comparison of IRPCA, LatLRR, PNMF and SPLRR vs. training numbers on: (a) HWDB1.1-D database, (b) MNIST database.

Table 2: Performance comparison of IRPCA, LatLRR, PNMF and SPLRR on each database.

Result / Method	**HWDB1.1-D**		**MNIST digits**	
	Mean	Best	Mean	Best
IRPCA	87.95%	91.06%	87.54%	90.79%
LatLRR	87.56%	90.99%	87.89%	91.05%
PNMF	87.99%	90.50%	86.45%	90.04%
SPLRR	89.38%	91.70%	88.27%	91.24%

For HWDB1.1-D, we test each method through varying the number of training samples per digit from 20 to 160 with step 20. For MNIST, we compare the performance of each method by varying the training number of each digit from 30 to 240 with step 30. For each training number, results are averaged over 15 runs. The results on the databases are plotted in Figure 5. The mean and best results according to Figure 5 are given in Table 2. The following observations can be found. First, from Figure 5, our proposed SPLRR outperforms all its competitors across all training numbers in most cases. The advantages of SPLRR lie in its ability to automatically extract visual saliency features from the corrupted data. IRPCA obtains comparable results to LatLRR for each case. The overall performance of PNMF is better than IRPCA and LatLRR on the HWDB1.1-D dataset, whilst its results are inferior to them on the MNIST dataset. Second, similar findings are obtained from Table 2.

V. Concluding Remarks

In this paper, we have presented a Sparse Projection and Low-Rank Recovery (SPLRR) framework. For high-dimensional data representation, SPLRR clearly considers the sparsity and the low-rankness properties. SPLRR is originally proposed for handwritten representation and recognition. SPLRR can do handwriting repairing, learn a sparse projection to extract the salient stroke features and simultaneously remove the possible errors. For image recovery, our SPLRR can effectively handle the corruptions, repair the missing pixel values and correct the irregular strokes of the handwritings. For feature extraction, SPLRR is able to extract the visual saliency features from the training samples, which can best distinguish handwritings in classification problem. For handwritten digits recognition, we experimentally observe that the sparse projection of SPLRR is as much powerful as the low-rank projection of some existing related methods for feature extraction and recognition. Note that SPLRR cannot reduce the dimensionality of data, thus an important future direction is to consider SPLRR or its sparse and low-rank outputs for subspace learning [11][13][38].

Acknowledgement

The work described in this paper was partially supported by the Natural Science Foundation of Jiangsu Province under Grant No. SBK201222725.

References

[1] A. Ganesh, J. Wright, X.D. Li, E. Candes, Y. Ma. Dense Error Correction for Low-Rank Matrices via Principal Component Pursuit. In: *Proceedings of the International Symposium on Information Theory (ISIT)*, 2010.

[2] E. Candes, and P. Yaniv. Matrix completion with noise. In: *Proceedings of the IEEE*, 98(6):925-936, 2009.

[3] Z.C. Lin, A. Ganesh, J. Wright, L.Q. Wu, M.M. Chen, and Y Ma. Fast Convex Optimization Algorithms for Exact Recovery of a Corrupted Low-Rank Matrix. *UIUC Technical Report UILU-ENG-09-2214*, 2009.

[4] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In: *Proc. Neural Inf. Process. Syst.*, pp.1-9, 2009.

[5] T. Y. Zhou, and D. C. Tao. GoDec: Randomized Lowrank & Sparse Matrix Decomposition in Noisy Case. In: *Proceedings of*

the International Conference on Machine Learning (ICML), pp.33-40, 2011.

[6] G. C. Liu, Z.C. Lin, S.C. Yan, J. Sun, Y. Yu, and Y. Ma. Robust Recovery of Subspace Structures by Low-Rank Representation. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 85(1):663-670, 2012.

[7] W.T. Tan, G. Cheung, and Y. Ma. Face Recovery in Conference Video Streaming using Robust Principal Component Analysis. In: *Proceedings of IEEE International Conference on Image Processing*, 2011.

[8] E. Candes, X. D. Li, Y. Ma, and J. Wright. Robust Principal Component Analysis? *Journal of the ACM*, 58(3):1-37, 2011.

[9] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Technical report,* 2009.

[10] J. F. Cai, E. J. Cand`es, Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[11] Z. Zhang, M.B. Zhao, and T.W. S. Chow. Constrained Large Margin Local Projection Algorithms and Extensions for Multimodal Dimensionality Reduction. *Pattern Recognition*, 46(12):4466-4493, July 2012.

[12] B. K. Bao, G. C. Liu, C. S. Xu, and S. C. Yan. Inductive Robust Principal Component Analysis. *IEEE Trans. on Image Processing (TIP)*, vol.21, no.8, pp. 3794-3800, 2012.

[13] Z. Zhang, T. W. S, Chow, M..B. Zhao. M-Isomap: Orthogonal Constrained Marginal Isomap for Nonlinear Dimensionality Reduction. *IEEE Trans. on Systems, Man and Cybernetics Part B: Cybernetics,* 43(1):180-192, Feb 2013.

[14] Y. L. Cun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278-2324, 1998.

[15] Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. TILT: Transform invariant low-rank textures. In: *Proceedings of the Asian Conference on Computer Vision*, pp.314-328, 2010.

[16] Z. Zhang, M.B. Zhao, and T. W. S. Chow. Binary- and Multi-Class Group Sparse Canonical Correlation Analysis for Feature Extraction and Classification. *IEEE Trans. on Knowledge and Data Engineering*, 2012. To appear.

[17] J. Wright, A. Yang, S. Sastry, Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(2):210-227, 2009.

[18] M. Zheng, J. J. Bu, C. Chen, C. Wang, L. J. Zhang, G. Qiu, and D. Cai. Graph Regularized Sparse Coding for Image Representation. *IEEE Trans. on Image Processing*, 20(5): 1327-1336, 2011.

[19] L. S. Qiao, S. C. Chen, and X. Y. Tan. Sparsity Preserving Projections with Applications to Face Recognition. *Pattern Recognition*, 43(1):331-341, 2010.

[20] S. Gao, I. Tsang, and L. Chia. Laplacian Sparse Coding, Hypergraph Laplacian Sparse Coding, and Applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(1):92-104, 2013.

[21] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 2004.

[22] Y.G. Peng, A. Ganesh, J. Wright, W.L Xu, and Y. Ma.RASL: Robust Alignment by Sparse and Low-Rank Decomposition for Linearly Correlated Images. *IEEE Trans. on Pattern Analysis and Machine Intelligence,*34(11):2233-2246, 2012.

[23] I. Jolliffe. Principal Component Analysis. Springer-Verlag, 1986.

[24] Z. R. Yang, and E. Oja. Linear and Nonlinear Projective Nonnegative Matrix Factorization. *IEEE Trans. on Neural Networks*, 21(5):734-749, 2010.

[25] Z. Zhang, T. W. S, Chow, M.B. Zhao. M-Isomap: Orthogonal Constrained Marginal Isomap for Nonlinear Dimensionality Reduction. *IEEE Trans. on Systems, Man and Cybernetics Part B: Cybernetics*, vol.43, iss.1, pp.180-192, Feb 2013.

[26] F. Yin, C. and L. Liu. Handwritten Chinese text line segmentation by clustering with distance metric learning. *Pattern Recognition*, 42:3146-3157, 2009.

[27] U. Y. Nahm, M. Bilenko, R.J. Mooney. Two approaches to handling noisy variation in text mining. In: *Proceedings of the ICML-2002 Workshop on Text Learning*. Sydney, Australia, 2002.

[28] S.T. Yuan, and J. Sun. Ontology-based structured cosine similarity in document summarization: with applications to mobile audio-based knowledge management. *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, 35(5):1028-1040, 2005.

[29] C. L. Liu, F. Yin, D.H. Wang, Q. F. Wang.Online and offline handwritten Chinese character recognition: Benchmarking on new databases. *Pattern Recognition*, 46(1):155-162, 2013.

[30] C. W. Hsu, C. C. Chang, and C. J. Lin. *A Practical Guide to Support Vector Classification*. 2010.

[31] G. C. Liu, and S. C. Yan. Latent Low-Rank Representation for subspace segmentation and feature extraction. In: *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[32] M. Aharon, M. Elad, A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54(11): 4311-4322, 2006.

[33] M. Bulacu, and L. Schomaker. Text-Independent Writer Identification and Verification Using Textural and Allographic Features. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(4):707-717, 2007.

[34] X. F. He, S. C. Yan, Y. X. Hu, P. Niyogi, and H. J. Zhang. Face Recognition Using Laplacianfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(3):328-340, 2005

[35] E. Elhamifar, and R. Vidal. Sparse subspace clustering. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol.2, pp.2790-2797, 2009.

[36] G. Zhu, S. Yan, and Y. Ma. Image tag refinement toward low-rank, content-tag prior and error sparsity. In: *Proc. Int. Conf. Multimedia*, pp. 461-470, 2010.

[37] K. Min, Z. Zhang, J. Wright, and Y. Ma. Decomposing background topics from keywords by principal component pursuit. in *Proc. 19th Int. Conf. Inf. Knowl. Manage.*, pp. 269-278, 2010.

[38] M. B. Zhao, Z. Zhang, T. W. S. Chow. Trace Ratio Criterion based Generalized Discriminative Learning for Semi-Supervised Dimensionality Reduction. *Pattern Recognition*, 45(4):1482-1499, April 2012.