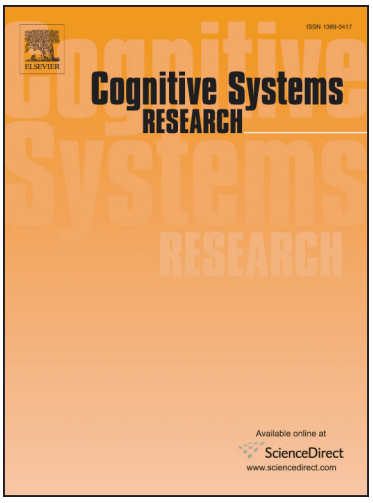
Journal Pre-proofs



Robust supervised sparse representation for face recognition

Jian-Xun Mi, Yueru Sun, Jia Lu

PII: S1389-0417(20)30010-3
DOI: https://doi.org/10.1016/j.cogsys.2020.02.001
Reference: COGSYS 925

To appear in: *Cognitive Systems Research*

Revised Date: 12 August 2019
Accepted Date: 15 February 2020

Please cite this article as: Mi, J-X., Sun, Y., Lu, J., Robust supervised sparse representation for face recognition, *Cognitive Systems Research* (2020), doi: https://doi.org/10.1016/j.cogsys.2020.02.001

Robust supervised sparse representation for face recognition

Jian-Xun Mi[a,b], Yueru Sun[a], Jia Lu[c]

[a] *Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[b] *College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China*
[c] *College of Computer and Information Sciences, Chongqing Normal University, Chongqing, 401331, China*

Abstract

Sparse representation based classification (SRC) has become a popular methodology in face recognition in recent years. One widely used manner is to enforce minimum l_1-norm on coding coefficient vector, which is considered as an unsupervised sparsity constraint and usually requires high computational cost. On the other hand, supervised sparsity representation based method (SSR) realizes sparse representation classification with higher efficiency by multiple phases of representing a probe. Nevertheless, since previous SSR methods only deal with Gaussian noise, they cannot satisfy empirical face recognition application which faces wide variations. In this paper, we propose a robust supervised sparse representation (RSSR) model, which uses two-phase of robust representation to compute a sparse coding vector. Huber loss is employed as the fidelity term in the linear representation, which improves the competitiveness of correct class in the first phase. Then training samples with weak competitiveness are removed by supervised way. In the second phase, the competitiveness of correct class is further boosted by Huber loss. We compare the RSSR with other state-of-the-art methods under different conditions, including illumination variations, gesture changes, expressions, corruptions, and occlusions. Comprehensive ex-

*Corresponding author
Email address: mijianxun@gmail.com (Jian-Xun Mi)

periments on four open databases demonstrate the robustness of RSSR and competitive performance is obtained in dealing with face images with occlusion or not.

Keywords: face recognition, huber Loss, supervised sparse representation

1. introduction

Using biometric identification technology for verifying identity is gaining more importance, and different techniques have been developed such as Palmprint recognition[1][2] and face recognition(FR)[3][4]. FR[5][6][7]has been extensively studied for its broad application prospects in recent years, such as authentication and payment system. The primary task of FR consists of feature extraction and classification[8][9][10]. For many classifiers, feature extraction that tends to discover discriminative feature is very important, which has great influence on recognition rate. Since there is rich redundancy in a face image, low dimensional features are extracted to concisely represent the samples in training set[11][12][13][14]. So that using these features can alleviate the computational cost and improve the recognition performance of classifiers. For empirical applications of FR, various changes including lighting, expression, pose, and occlusion can be seen in a probe and which could not included in training set, which leads to the consequence that the computed feature becomes inefficient. Image segmentation can be used to help the effective representation of an image by retaining the informative parts of images[15][16]. Some methods based on neural networks[17][18][19] can achieve good classification. However, these methods always require a large number of training samples[20], and there many unexplainable hyperparameters in the neural works[21]. Furthermore, some studies believe that feature extraction methods, such as principle component analysis (PCA)[8][22], linear discriminate analysis (LDA)[4][23], and independent component analysis (ICA)[24], have low effect on classifications, which are built based on linear representation technique[14]. In this way, the key importance of a FR system is to develop a classifier which makes the most

use of the discriminative information in a probe in variation and/or corruption conditions[25][26].

To this end, methods based on linear regression (LR) are proposed recently which show some promising results in FR[27][28][29][30]. The linear regression based classification (LRC) is a typical method which represents a probe one class at a time[31]. The robust version of LRC is proposed in [32] which shows excellent performance in dealing with illumination change. The sparse representation based classification (SRC) is the first attempt to introduce sparse representation into FR research area which is able to enhance the discrimination and robustness of classification[33]. The follow-up works, including correntropy-based sparse representation (CESR)[34] and regularized robust coding with l_1-norm (RRC1)[35]. The CESR uses the conrrentropy-based Gaussian kernel function as fidelity term, which has unsatisfactory performance in the case of scarf occlusion and illumination change. The regularized robust coding (RRC)[35] is a complex model which has some parameters to adjust as to obtain good performance under different situations. To implement sparse representation differently, Xu et al. proposed a method called two-phase test sample sparse representation (TPTSR)[36]. Besides, two similar works are proposed in [37][38] which use different schemes to conduct supervised sparse representation, and some works use $l_{2,1}$-norm as penalty term combined with LDA[39].

The basic assumption behind the aforementioned methods is that a probe can be linearly represented by training samples from the same class of the probe[40][41]. There are two critical points for these methods. First, since the match error determines the final decision, mitigating the impact of corrupted pixels of the probe can eliminate possible bias result from the corruption. Second, the model parameter, i.e. coefficient vector, should be properly estimated in order to enhance discriminative capability of models. The first one has to do with the fidelity term. If l_2-norm is used as a fidelity term, the large values of match error caused by pixel corruption or occlusion have great impact on the fidelity term. When the number of these distorted pixels is large, the linear regression based method may fail. In this case, robust estimation function works

better as fidelity term [42][43][44][45], since the impact of overly distorted pixels is restrained. According to the second critical point, we can categorize linear regression based methods into two groups. One is to estimate the coefficient vector one class by one class[31]. Another one is to represent a probe collaboratively on the whole training set so that all coefficient vectors are estimated at one time. In [27][33], it is suggested that the collaborative representation strategy is helpful to improve recognition accuracy, in this way the sample space is more complete and a probe can be represented better. In [33], the concept of sparse collaborative representation is proposed, and a probe is collaboratively represented while the coefficient vector is subject to some sparseness constraint, which increases the discriminative power of the regression

In this paper, we work towards the direction of developing a novel linear regression based classification method mainly considering the above two points. Hence a robust supervised sparse representation (RSSR) method is proposed for FR. The contributions of RSSR method are outlined as follows:

- RSSR method employs Huber loss as the fidelity term in the linear representation. As the same as the squared loss, Huber loss is a minimum-variance estimator of the mean. However, Huber loss is superior to l_2-norm based loss as a fidelity term when there are large outliers.

- RSSR uses two-phase representation scheme to implement supervised sparse representation. The first phase, referred to as the coarse representation, uses all training samples to represent a probe; while in the second phase, referred to as a fine supervised sparse representation, only training samples which have high contribution in the coarse representation are used and the coefficients for the rest training samples are set to zero.

The rest of the paper is organized as follows: Firstly, we briefly review some existing relevant methods which use linear representation for robust FR and discuss their limitations in the next section. In Section 3, we present the RSSR model and the corresponding classification algorithm. The further discuss of the robustness and sparsity of the RSSR is shown in Section 4. In Section 5, we

compare the RSSR with other related state-of-the-art methods by conducting extensive experiments on different public available face databases.

2. Related works

To present a sparse representation face recognition scenario, we assume that a test image $\mathbf{y}$ can be approximately represented by the gallery of train images $\mathbf{A}$ i.e.

$$\mathbf{y} \approx \mathbf{A}\mathbf{x} \tag{1}$$

where $\mathbf{x}$ is a coefficient vector. According to previous studies[33], it is safely to assume that a test sample can be represented by the training samples from the same class at least. Hence by conducting collaborative representation as Eq.(1), a likely solution of $\mathbf{x}$ is sparse because most entries equal zeros except the entries associating with training samples form correct class. The collaborative representation problem can be recast as a sparse representation issue. When a test sample contains noises or variations, the solution of $\mathbf{x}$ becomes no longer sparse. To address this problem, methods have been proposed so far to add sparse constraint on $\mathbf{x}$ in linear representation procedure. These methods could be divided into two groups in terms of the way to implement sparsity: unsupervised sparse representation (USR) and supervised sparse representation (SSR).

2.1. Unsupervised sparse representation

Unsupervised sparse representation(USR) intends to incorporate collaborative representation and sparse representation together into a single optimization. Inspired by theory of compressed sensing[46][47][48], USR formulizes sparse representation as the following optimization problem:

$$\mathbf{x}^* = \arg\min \|\mathbf{x}\|_0 \quad s.t.\ \mathbf{y} = \mathbf{A}\mathbf{x} \tag{2}$$

where $\|\cdot\|_0$ denotes the l_0-norm of vector $\mathbf{x}$ which counts the number of nonzero entries in a vector. To directly solve Eq.(2) is a NP-hard problem, so that

alternative approaches are used to approximate the sparse solution. Fortunately, the l_1-norm minimization problem is equal to the l_0-norm minimization problem under certain condition[47][48]. Therefore, the sparse representation vector is computed by

$$\mathbf{x}^* = \arg\min \|\mathbf{x}\|_1 \quad s.t.\ \mathbf{y} = \mathbf{A}\mathbf{x} \tag{3}$$

where $||\mathbf{x}||_1 = \sum ||x_i||$, $\mathbf{x} = [x_1, x_2, ..., x_n]^T$. Further, Eq.(3) can be transformed into the follow equivalent form:

$$\mathbf{x}^* = \arg\min g(\mathbf{e}) + \lambda\|\mathbf{x}\|_1 \tag{4}$$

where $\mathbf{e} = \mathbf{y} - \mathbf{A}\mathbf{x} = [e_1, e_2, \cdots, e_m]$ which is called the match error vector, and $g(\cdot)$ is an error function which uses the squared loss $\|\cdot\|_2$, and λ is a small positive constant which balances the loss function and the regularization term. The above optimization problem Eq.(4) is also known as Lasso regression[49][50]. By solving Eq.(4), the SRC uses the sparse representation vector $\mathbf{x}^*$ to select which class has the most significant contribution of representing the probe. According to [33], this method obtains promising results in several FR scenarios. To further improve the performance of SRC, some follow-up studies suggest replacing the squared loss function in Eq.(4) with robust estimation function because the squared loss function could be affected severely by large entries of error vector. Compared with the squared loss function, the robust estimation function assigns weight to each entry of error vector according to its value. For example, a Gaussian function is used as its weighting function in CESR and in RRC the weighting function is a logistic function. According to the robust statistical theory[51], these methods reduce the influence of large outliers. Although l_0-norm is replaced by l_1-norm in these sparse representation methods, to solve a l_1-norm problem is still computationally expensive.

2.2. Supervised sparse representation

Supervised sparse representation (SSR) uses collaborative representation itself to supervise the sparsity of representation vector. The linear regression

model of SSR uses l_2-norm to regularize representation vector:

$$\mathbf{x}^* = \arg\min \|\mathbf{e}\|_2 + \lambda\|\mathbf{x}\|_2 \tag{5}$$

In this way, the solution can be computed with much lower time cost than l_1-norm based sparse regression model. In TPTSR [36], which uses a two-phase representation scheme to recognize a probe. The solution of the first-phase, denoted by $\mathbf{x}_1^*$, severs as a supervisor to select a subset of training samples which contains $M \ll n$ samples associating with have the first M greatest contributions in representation. Then in the second-phase the probe is represented only on the selected subset. By this coarse-to-fine two-phase representation, a sparse representation vector is obtained since $n - M$ entries of this vector can be considered to be set to zeroes after the first phase. Compared to USR, SSR acquires a true sparse representation vector while that of USR has many non-zeroes although which may be very small.

3. Robust coding based supervised sparse representation

3.1. Robust regression based on Huber loss

In a practical face recognition system, one needs to deal with the probe image under non-ideal condition. Here, we consider two common non-ideal cases that test images with occlusion or corruption. In both cases, the representation model shown in Eq.(1) has to be transformed as:

$$\mathbf{e} = \bar{\mathbf{y}} - \mathbf{A}\mathbf{x} \tag{6}$$

Since a portion of pixels of the probe image is randomly distorted, the distribution of the matching residual $\mathbf{e}$ in Eq.(6) becomes heavy tailed. Therefore the loss function should be carefully chosen because only a few functions are feasible to deal with heavy-tailed distribution. For example, if the quadratic functions are used in this case to develop a regression model, which is known as ordinary least squares regression (OLSR), the regression result would be inefficient and biased. The reason is that the least squares predictions are dragged towards

the outliers. However, the absolute value function cannot perform minimum-variance estimation while the OLSR can. It is well-know that the smaller variance of the estimator achieves, the better the performance of the estimator should be. So an ideal function used in this estimation scenario should combine the two merits that the minimum-variance estimation (advantage of the quadratic loss function) and the median-unbiased estimation (advantage of the absolute value function). In statistics, the Huber loss is used in robust regression, which has the both merits. Huber loss is known as an unbiased estimator which remedies the disadvantage of square loss that the result has the tendency to be dominated by some unexpected contaminated features of a probe sample. In the even that a partial face image is occluded, the representation error shows a heavy-tailed distribution which influences the estimation of the representing accuracy and has an unpredictable effect on the classification stage. Fidelity term using Huber loss is much less sensitive to those outlying features and, at the same time, is less likely to miss minima, which occurs when absolute loss fidelity is used. Huber loss is defined as:

$$\mathrm{g}(e) = \begin{cases} \frac{1}{2}e^2 & \text{if} |e| \leq k \\ k|e| - \frac{1}{2}k^2 & \text{if} |e| > k \end{cases} \tag{7}$$

where k is a constant. As defined, Huber loss is a parabola in the vicinity of zero, and increases linearly above a given level $|e| > k$. In other words, Huber loss is able to tolerate the residuals with great absolute values, caused by unexpected pixel corruptions, which is superior to the quadratic loss function. Moreover, Huber loss has the quadratic curve which its input variable is very small. So that this estimator has the capability to correctly measure the small match error between the test and the prediction in corresponding to the uncontaminated pixels of the test.

Therefore, we develop a linear representation-based FR method using Huber loss as the error measure function, which is defined as:

$$G(\mathbf{e}) = \sum_i \mathrm{g}(e_i) \tag{8}$$

where g(.) is Huber loss, $e_i = y_i - d_i\mathbf{x}$, $y_i \in \mathbf{y}$, d_i is the i^{th} row of matrix $\mathbf{A}$. Unlike the OLSR problem, there is no close form solution for the regression problem using Huber loss. However, Eq.(8) can be recast as a weighted least squares regression form[52]. Here, we denote the first order derivative function of the Huber loss as $\psi(p) = \frac{dg(p)}{dp}$. So Eq.(8) becomes:

$$\sum \psi(e_i)\frac{\partial e_i}{\partial \mathbf{x}} = 0 \tag{9}$$

where $\psi(.)$ is called the influence function[51][52]. The influence function describes the extent that how match error affects the cost function and accordingly influences the estimation of coefficient vector. Next, we can define the weight function:

$$\omega(p) = \frac{\psi(p)}{p} \tag{10}$$

which assigns a specific weight to a certain pixel. The weight function of Huber loss is given by:

$$\omega(e_i) = \begin{cases} 1 & |e_i| \le k \\ 1/|e_i| & |e_i| > k \end{cases} \tag{11}$$

It is easy to see that if a pixel has a bigger value of residual (i.e.$|e_i| > k$) it will be assigned with a smaller weight (i.e.$1/|e_i|$). Although Huber loss and the l_1-norm fidelity employ a similar strategy (i.e. using same weighting function) to lower the influence of outliers in a probe, their weighting functions still have large difference which is shown in Fig.1. From Fig.1, one can see that the assigned weight by the l_1-norm fidelity can be infinity when the residual approaches to zero, making the coding unstable. However, for small residual (i.e.$|e_i| \le k$), Huber loss views it as thermal noise, and its weighting strategy is the same as the l_2-norm fidelity, i.e. $\omega(e_i) = 1$. Hence, the estimated coefficient vector by Huber loss is more stable. In addition, Huber loss follows the i.i.d GaussianLaplacian distribution, and [53] provides more theoretical persuasion. Integrating Eq.(8), Eq.(9), Eq.(10), and Eq.(11), Huber loss is able to be transformed to the regression form of iterated reweighted-least-squares as:

$$\min \sum_i w_{ii}{e_i}^2 \tag{12}$$

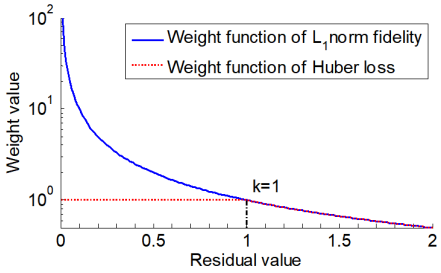


Fig. 1: Weight functions of the l_1-norm fidelity and Huber loss.

where w_{ii} is the weight of i^{th} pixel in $\bar{\mathbf{y}}$ which is obtained by Eq.(11). On the one hand, if we assume that outlier pixels have errors greater than k, the weight of these pixels should be as small weight as possible to ensure the estimator with median-unbiased. On the other hand, the estimator has an inherent advantage that it can achieve minimum-variance estimation of coefficient vector for clean pixels. There are many optional M-estimators, which have some good features to deal with outliers. If an M-estimator can match the distribution of errors, the coefficient vector can be estimated precisely[35]. However, face images are commonly involved in rich variations and unpredictable noises, which lead to a inscrutable distribution of $\mathbf{e}$. Considering the comprehensive performance of a FR method, Huber loss is more suitable for robustly estimating residuals.

3.2. The Proposed Model

In this paper, the Huber loss is used to develop a supervised sparse representation model, which needs two phases of representing a probe. In the first phase, a probe is represented collaboratively and the corresponding coding vector is computed by solving the following problem:

$$\mathbf{x} = \arg\min_{\mathbf{x}} \; G(\mathbf{y} - \mathbf{A}\mathbf{x}) + \tfrac{\lambda}{2}||\mathbf{x}||_2^2 \tag{13}$$

where $G(\mathbf{y}-\mathbf{A}\mathbf{x}) = G(\mathbf{e}) = \sum_{i=1}^{m} \mathrm{g}(e_i)$ and $\mathrm{g}(.)$ is the Huber loss defined as Eq.(7). λ is a parameter, which balances the fidelity term and the regularization term. Since the Huber loss is a piecewise function, we need to transform Eq.(13),

so as to facilitate the computation[54]. First of all, we define a sign vector $\mathbf{s} = [s_1, s_2, \cdots, s_j, \cdots, s_m]^T$ by

$$s_j = \begin{cases} -1 & \text{if} \quad e_j < -k \\ 0 & \text{if} \quad |e_j| \leq k \\ 1 & \text{if} \quad e_j > k \end{cases} \tag{14}$$

and define a diagonal matrix $\mathbf{W} \in {}^{m \times m}$ with diagonal elements

$$w_{jj} = 1 - {s_j}^2 \tag{15}$$

Now,by introducing $\mathbf{W}$ and $\mathbf{s}$, $G(\mathbf{e})$ can be rewritten as

$$G(\mathbf{e}, \mathbf{s}, \mathbf{W}) = \tfrac{1}{2}\mathbf{e}^{\mathrm{T}}\mathbf{W}\mathbf{e} + k\mathbf{s}^{\mathrm{T}}(\mathbf{e} - \tfrac{1}{2}k\mathbf{s}) \tag{16}$$

If all residuals are small (their values are smaller than k), the second term of the right side of Eq.(16) will disappear and the fidelity term degrades to quadratic function. Actually, in Eq.(16), the entries of $\mathbf{s}$ can be considered as indicators of large residuals, which divide the residuals into two groups, i.e. The small residuals and the large ones. These two groups of residuals are handled by first term and the second term of the right side of Eq.(16) respectively. The second term of the right side of Eq.(16) actually plays the same role as l_1-norm, but with a different variant. By substituting the fidelity term of Eq.(13) with Eq.(16), the optimization function now is reformed as

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \ \tfrac{1}{2}\mathbf{e}^{\mathrm{T}}\mathbf{W}\mathbf{e} + k\mathbf{s}^{\mathrm{T}}(\mathbf{e} - \tfrac{1}{2}k\mathbf{s}) + \tfrac{\lambda}{2}||\mathbf{x}||_2^2 \tag{17}$$

where $\mathbf{s}$ and $\mathbf{W}$ are defined in Eq.(14) and Eq.(15) respectively, and $\widehat{\mathbf{x}}$ is estimated coding vector. Each entry in $\widehat{\mathbf{x}}$ represents the contribution of the corresponding training sample for representing the test sample, i.e. the large absolute value of an entry means great influence in the representation. It is very common that some contaminated pixels exist in a test image, which results in more dense representation due to the compensation of the contaminated pixels. It is well known that sparse representation is helpful to improve the recognition accuracy. The concept of supervised sparse representation is that the training

samples without significant contribution in the first stage should be removed, so as to enforcing the sparsity of the coding vector.

To implement supervised sparse representation, we need to remove the training samples with low contribution from the next phase representation. We notice the fact that some entries in $\widehat{\mathbf{x}}$ are equal or closed to zeros, which means the corresponding training samples have very low contribution to represent the test sample. Hence these training samples are to be removed. Here, we assume there are M training samples, which have the M greatest contribution, are retained after the first phase of representation. Hence the training sample set becomes $\tilde{\mathbf{A}} \in \mathbb{R}^{m \times M}$($\tilde{\mathbf{A}}$ is a subset of $\mathbf{A}$ and $M \ll n$).

In the second phase, the test sample is represented anew over the retained M candidate samples,i.e. $\mathbf{y} \approx \tilde{\mathbf{A}}\mathbf{x}'$. Due to this second phase of representation involves less gallery samples, noises in test sample are less likely to be compensated. It is beneficial for Huber loss to find polluted pixels. Moreover, the final Huber loss $\widehat{\mathbf{x}}^{'} \in \mathbb{R}^{M}$ of the coding vector can be obtained by Eq.(17). Due to the interferential samples are removed after the first phase, the correct samples, training samples belong to the same class as a test sample, tend to have the greater contributions to represent the test sample in second phase[36]. Therefore, the coefficients of correct samples are highlighted significantly in $\widehat{\mathbf{x}}^{'}$.

3.3. Algorithm of RSSR

In both two phases of representation, we need to solve the robust regression optimization problem as given in Eq.(17). However, this optimization objective function has no close form solution. In order to improving the optimization efficiency of RSSR, two extra variables $\mathbf{s}$ and $\mathbf{W}$ are introduced into the objective function. That is, $\mathbf{s}$ and $\mathbf{W}$ make the function and its derivative to become more concise. Therefore, the objective function Eq.(17) can be transformed to following:

$$\mathbf{A}^T\mathbf{W}\mathbf{A}\mathbf{x} - \mathbf{A}^T\mathbf{W}\mathbf{y} + k\mathbf{A}^T\mathbf{s} + \lambda\mathbf{x} = 0 \tag{18}$$

The coefficient vector is represented by the follow formula:

$$\mathbf{x} = (\mathbf{A}^T\mathbf{W}\mathbf{A} + \lambda\mathbf{I})^{-1}(\mathbf{A}^T\mathbf{W}\mathbf{y} - k\mathbf{A}^T\mathbf{s}) \tag{19}$$

where $\mathbf{I}$ is an identity matrix. In this paper, the final coding vector $\mathbf{x}$ is solved the by the iteratively re-weighting technique. The positive-weight matrix $\mathbf{W}$ and the passive-weight vector $\mathbf{s}$ can be computed by the following formula

$$\mathbf{s}^t = \mathbf{s}(\mathbf{x}^{t-1}) \tag{20}$$

$$\mathbf{W}^t = \mathbf{W}(\mathbf{s}^t) \tag{21}$$

where t is the number of iteration, and $\mathbf{s}(.)$ and $\mathbf{W}(.)$ represent Eq.(14) and Eq.(15), respectively. In this paper, the initial vector $\mathbf{x}^0$ uses the result of Ridge regression, i.e. $\mathbf{x}^0 = (\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^T\mathbf{y}$. And the formula of the coding vector $\mathbf{x}$ with t^{th} iteration as

$$\mathbf{x}^t = \mathbf{x}^{t-1} + u^t \mathrm{h}(\mathbf{W}^t, \mathbf{s}^t) \tag{22}$$

where $h(\mathbf{W}^t, \mathbf{s}^t) = (\mathbf{A}^T\mathbf{W}^t\mathbf{A} + \lambda\mathbf{I})^{-1}(\mathbf{A}^T\mathbf{W}^t\mathbf{y} - k\mathbf{A}^T\mathbf{s}^t) - \mathbf{x}^{t-1}$, and $0 < u^t \leq 1$, which is a step size that makes the loss value of the t-th iteration less than that of the last. In this paper, u^t is determined via the golden section search if $t > 1$ ($u^1 = 1$). The detailed estimated procedure of coding vector of RSSR is summarized in the Algorithm 1.

After computing the coding vector of a test sample, we use a typical classification rule for collaborative representation to make the decision. The test sample is classified by the following strategy:

$$indentity(\mathbf{y}) = arg\min_c d_c \tag{23}$$

And

$$d_c = G(\mathbf{y} - \tilde{\mathbf{A}}_c\hat{\mathbf{x}}_c, \hat{\mathbf{s}}, \hat{\mathbf{W}}) \Big/ ||\hat{\mathbf{x}}_c||_2 \tag{24}$$

where $\tilde{\mathbf{A}}_c$ is a sub-dictionary that contains samples of class c, and $\mathbf{x}_c$ is the associated coding coefficient vector of class c.

3.4. Time Complexity of the proposed Algorithm

The main computational consumption of RSSR is spent on calculating of the coding vector, by Eq.(19). To simplify the analysis of time complexity, we

Algorithm 1

Input: test image , and training samples $\mathbf{A}$

Output: $\hat{\mathbf{x}}$

repeat

  Compute residual $\mathbf{e}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t$

  Update $\mathbf{s}^t$, $\mathbf{W}^t$ and $\mathbf{x}^t$ by Eq.(20), Eq.(21) and Eq.(22), respectively.

until maximum iterations or convergence.

Selecting candidate samples

repeat

  Compute $\hat{\mathbf{x}}^l = (\tilde{\mathbf{A}}^T\tilde{\mathbf{A}} + \lambda\tilde{\mathbf{I}})^{-1}\tilde{\mathbf{A}}^T\mathbf{y}$

  Compute residual $\hat{\mathbf{e}}^l = \mathbf{y} - \tilde{\mathbf{A}}\hat{\mathbf{x}}^l$

  Update $\hat{\mathbf{s}}^l$,$\hat{\mathbf{W}}^l$,and $\hat{\mathbf{x}}^l$ via Eq.(20) , Eq.(21), and Eq.(22), respectively.

until maximum iterations or convergence.

transform it into

$$(\mathbf{A}^T\mathbf{W}\mathbf{A} + \lambda\mathbf{I})\mathbf{x} = (\mathbf{A}^T\mathbf{W}\mathbf{y} - k\mathbf{A}^T\mathbf{s}) \tag{25}$$

Note that $\mathbf{W}$ only contains two kinds of elements, i.e. 0 and 1. Therefore, the Eq.(22) can be converted into

$$(\mathbf{A_W}^T\mathbf{A_W} + \lambda\mathbf{I})\mathbf{x} = (\mathbf{A_W}^T\mathbf{y_W} - k\mathbf{A}^T\mathbf{s}) \tag{26}$$

The solution of Eq.(22) can be obtained by conjugate gradient method, whose time complexity is $\mathrm{O}(km_wn)$ [55], where k is the iteration number in conjugate gradient method, m_w is the dimensionality of face feature, and n is the number of training samples. The main time complexity of the coding process in the first phase is about $\mathrm{O}(t_1k_1m_wn)$ if the algorithm needs t_1 iterations to converge. Analogously, the time complexity is about $\mathrm{O}(t_2k_2m_wM)$ in the second phase of RSSR. Due to $M \ll n$ and t_1 is usual less than 10, the final time complexity of our algorithm is $\mathrm{O}(m_wn)$. In addition, the time complexity of our methods is less than RRC1 (i.e.$\mathrm{O}(m^2n)$) and regularized robust coding with l_2-norm (RRC2) [35] (i.e.$\mathrm{O}(mn)$) due to m_w is usually less than m.

4. Sparsity and robustness of our model

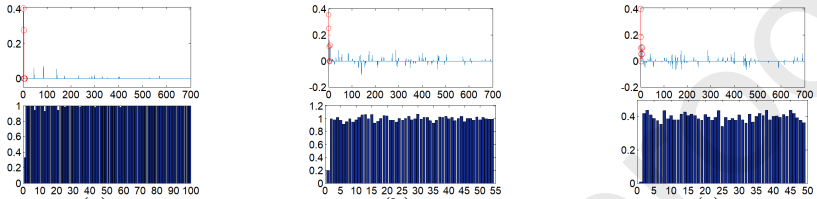
In this section, we analyze the effectiveness of the two terms in object function, i.e. the fidelity term and the regularization term, of RSSR when dealing with distorted facial images. Our aim is to demonstrate that the robustness and the sparseness of RSSR with comparison to SRC and TPTSR methods. Here, we adopt examples that two probe faces, i.e. one is clean and another is contaminated with 30% random pixel corruption. Both images are used as tests to compare three methods which are shown in Fig.2(a) and Fig.2(b) respectively. And the experiment setting on AR dataset is the same as[33], i.e. facial images without occlusion in session 1 are used for training and test images belongs to session 2. More experiments were reported in section 5. As

(a) (b)

Fig. 2: (a)A face image from the first class. (b)The corrupted test sample of AR data base.

we known, linear representation based FR methods have an assumption that the test sample $\mathbf{y}$ and these samples $\mathbf{A}^i$ of the i-th class should lie in the same subspace if $\mathbf{y}$ belongs to the i-th class[31]. Therefore, the training samples from the correct class can provide a compact representation of the test sample. In Fig.3, three sparse representation based classifiers, including SRC, TPTSR, and the proposed RSSR, show very similar results that the samples from the correct class have great coefficient values, which means the test sample can be well represented by only a few training samples. If we consider the sparse representation as a competition among training samples, correct samples will make the major contribution in the representation. According to the typical decision rule for these linear representation based approaches, a test sample is very likely

to be classified to the class whose training samples have high representation contributions[33]. From Fig.3, Fig.4, and Fig.5, we can observe clearly that the representation residuals in RSSR are smaller than SRC and TPTSR, because the classifier in RSSR uses Huber loss rather than l_2-norm in SRC and TPTSR to calculate the distance between the test image and a certain class.



(a) (b) (c)

Fig. 3: (a)-(c) Top: the representation coefficients for SRC, TPTSR, and RSSR in which the coefficients associating with the training samples from class 1 are marked with red, Bottom: the corresponding representation residuals (for TPTSR and RSSR, classes which have none training samples involved in the second representation step are not shown).

In practice, empirical application of a face recognition system should tolerate some common variations. Here the noise caused by pixel corruption is used as an example. The test sample $\mathbf{y}'$ (Fig.2 (b)) we used now is the same one as Fig.2(a) but with pixel corrupted up to 30% and we compare the representation results with same experiment set as Fig.3. In the image feature, the noisy probe $\mathbf{y}'$ can deviate greatly from $\mathbf{y}$, so that the spatial relation between $\mathbf{y}$ and $\mathbf{A}^1$ can be distorted, which leads to a bad representation of $\mathbf{y}'$ by the correct class $\mathbf{A}^1$. That is to say, distorted samples violate the mentioned assumption of linear representation. It is the key to employ good residual measurement to reduce the influence brought by noises. The previews TPTSR and SRC use Euclidean distance $\|.\|_2$ to measure the similarity between two images, which is very sensitive to pixel corruption. In our method, Hubers loss function suppresses the distortion caused by corrupted pixels to distance measurement. Let us check the comparison shown in Fig.4: For SRC, with the help of l_1-norm restriction on the coefficient vector, the representation is very sparse, which is helpful to tolerate noises . But the samples from class 1 are not only ones which

have high representation contribution and the training sample having the highest contribution is from class 31 so that the noisy probe could be misclassified. For TPTSR, since no mechanism is used to deal with the strong noise, training samples from class 1 are not competitive at all. Actually TPTSR is designed to perform highly efficient supervised sparse representation but does not consider serious variations. For our RSSR, the training samples associating with very large coefficients all comes from class 1 which means $\mathbf{y}'$ can be identified correctly. In other words, we make the supervised sparse representation possible to recognize a very noisy probe.

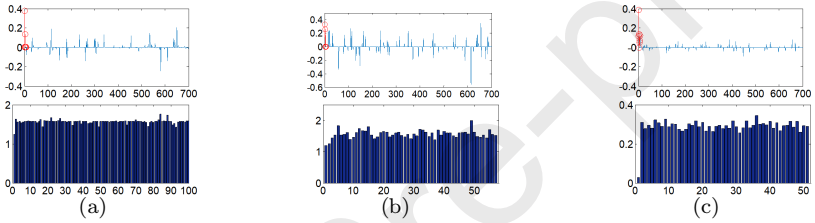


Fig. 4: The representation results for corrupted image shown in Fig. 2 (b). (a)-(c) Top: the representation coefficients for SRC, TPTSR, and RSSR in which the coefficients associating with the training samples from class 1 are marked with red; Bottom: the corresponding representation residuals (for TPTSR and RSSR, classes which have none training samples involved in the second representation step are not shown).

Next, we give a detailed demonstration of the efficiency of the proposed method in dealing with noisy image by comparing with the previous supervised sparse representation based method. We can see in Fig.5 that in the first phase of representing a probe the previous TPTSR has very dense coefficients while our RSSR produces an approximate sparse coefficient vector. More importantly, for RSSR samples from the correct class have the major contribution to represent the probe so that the residual bar of the first class is lower than that of others, however, for TPTSR no class shows significant competitiveness over others. It means the noises hinder collaboration representation used in both phases of TPTSR from sparsely coding the probe. As shown in Fig.4, even after removing some low contribution samples in the first phase, TPTSR still cannot highlight

the representation load of samples from correct class, which means by enforcing the supervised sparsity on coefficients TPTSR not able to class a noisy face. While we can see that RSSR further reduces the residual of correct class in the second phase, which mean in this noisy case supervised sparsity works well to improve the efficiency of representation. In summary, RSSR introduces Huber loss into supervised sparse representation based classification solves the problem that noisy face images could invalidate the previous TPTSR.

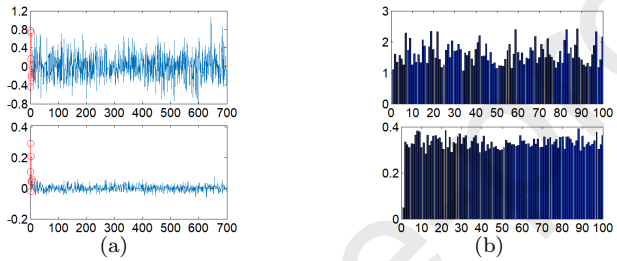


Fig. 5: Comparison of coding efficiency of two supervised sparse methods in the first phase of representation (i.e. top: TPTSR and bottom: RSSR).

5. Experiment results

In this section, the performance of RSSR is evaluated via extensive experiments. We compare RSSR with six state-of-the-art methods, including SRC[33], CESR[34], RRC1[35], RRC2[35], CRC[27], LRC[31], and TPTSR (for TPTSR the candidate set is set to 10 percent of training set)[36], ProCRC[56], NMR[57]. And all compared methods except LRC are collaborative representation based, while LRC is to represent a probe one class at a time. All used gray-scale images are normalized to unit vectors. Experiments were implemented on MATLAB using a desktop with 3.30GHz Intel CPU and 16G RAM.

5.1. Parameter Setting

There are three parameters in RSSR, i.e. the Huber threshold k, the sparse factor M, and the Lagrangian multiplier λ. k is important to Huber loss which

determines which part of Huber loss is used to handle the regression error, i.e. the quadratic function or absolute value function. The empirical value of k can be obtained in many ways[58][59]. However, according to our extensive experiments, the value of k is set to 0.001. As discussed in[36][60], the supervised representation based methods have good performance when the sparse factor is in arrange from 0.1 to 0.2. Therefore, we set the parameter M to 0.1. For λ which is used to balance the fidelity term and regulation term, it is set to 0.001 in all the experiments. The value of λ is selected on the validation set by multiple iterations of cross-validation.

5.2. Face Recognition without occlusion

We first evaluate the performance with facial images with variations, such as illumination, posture changes and expression changes but without occlusion. And the experiments are performed on four public available face recognition databases, namely, AR[61], Extended Yale B[62][63], Feret[64], and ORL[52].

Extended Yale B Database. We used 31 subjects in the dataset each of which has 64 frontal facial images under various lighting conditions[62][63] According to different lighting intensities, the database can be divided into five subsets. The degree of lighting interference is increasing from subset 1 to subset 5. Images from the subset 1, normal-to-moderate lighting conditions (shown in Fig.6 top), were used for training samples and the subset 4 with more extreme lighting conditions (shown in Fig.6 bottom) was used for test. Table 1 shows that other robust regression based methods such as CESR, RRC1, and RRC2 perform poorly in dealing with illumination variations, and their corresponding recognition rates are only 16.1%, 68.4%, 63.8%, respectively. These methods enforce too strong constraint on some non-noise pixels by the weighting function, which destroys the illumination model. Although our proposed RSSR belongs to robust regression based method, it greatly outperforms the other three and the recognition rate reaches 87.6%. This is because Huber loss can assign a better weight to pixels with large residual values. In the illumination condition, the

coding vector might contains some the negative elements, but those samples with negative coefficients are very likely to have been removed in the first step of TPTSR, which leads to the poor performance of TPTSR that is 55.8%. The recognition rate of SRC only reaches 15.9%, which indicates that SRC cannot deal with too intense lightening changes. Moreover, LRC, ProCRC, and CRC obtain good recognition rate, i.e. 88.7%, 77.0% , and 85.9%, respectively.

Fig. 6: Some images from the Extended Yale B database. Top: sample images from subset 1. Bottom: sample images from subset 4.

Table 1: Results for the Extended Yale B database

Method	CESR	RRC1	RRC2	LRC	SRC	CRC	TPTSR	ProCRC	NMR	**RSSR**
Recognition rate(%)	16.1	68.4	63.8	88.7	15.9	85.9	55.8	77.0	43.8	**87.6**

AR Database. The AR database consists of more than 4000 frontal facial images from 126 subjects (70 men and 56 women)[61]. For each subject, 26 images were taken in two separate sessions (i.e. session 1 and session 2). And these images included different facial variations, such as expressions changes, illumination variations, and different disguises (sunglass and scarf). As in [35], we choose a subset from AR database which contained 50 male and 50 female subjects. For each individual, the used training set included first seven images from session 1 with expression variations and illumination variations (Fig.7 top), the other seven images from session 2 were used for test (Fig.7 bottom). In addition, all images were downsampled to 50×40 . The comparison results of the several methods are listed in Table 2. The recognition rates of LRC (76.1%), NMR (71.6%) and SRC (75.7%) are the lowest among all of methods. Besides SRC,

the recognition rates of these methods based on collaborative representation are over 90%. TPTSR (91.9%), CRC (92.7%), and CESR (91.3%) obtained similar results. RRC1 (96.7%), RRC2 (96.2%), and RSSR (95.0%) perform better.

Fig. 7: The part of images of one subject in AR. Top: these facial images from session 1. Bottom: these images from session 2.

Table 2: Results for the AR database

Method	CESR	RRC1	RRC2	LRC	SRC	CRC	TPTSR	ProCRC	NMR	**RSSR**
Recognition rate(%)	91.3	96.7	96.3	76.1	75.7	92.7	91.9	74.6	71.6	**95.0**

FERET Database. As in ([36]), a subset that contains 200 subjects were used from the FERET database[64]. Each individual had 7 images (i.e.Fig.8) from b series of FERET database whose names are marked with two-character strings: ba, bj, bk, be, bf, bd, and bg. We used three images: the first, third, and fourth facial of each subject, as training and the other four facial images were used as test. These images were downsampled to 40×40. From Table 3, we can see the recognition rate of RSSR (78.1%) is highest among all of methods. Note that that TPTSR (69.0%) and ProCRC (64.2%) outperform CRC (54.0%), which shows that in this case the supervised sparse constraint is conducive to classification. RRC1, CESR, and SRC both using l_1-norm constraint in their regularization obtain recognition rates up to 77.0%, 75.0%, and 73.4% respectively. RRC2 lags RRC1 by 14% for not using sparse constraint. In addition LRC obtains recognition rate of 72.6%.

Fig. 8: The part of images of the first subject in FERET.

Table 3: Results for the FERET database

Method	CESR	RRC1	RRC2	LRC	SRC	CRC	TPTSR	ProCRC	NMR	**RSSR**
Recognition rate(%)	75.0	77.0	63.0	72.6	73.4	54.0	69.0	64.2	63.3	**78.1**

ORL Database. ORL database contains 40 subjects and each individual provides 10 face images[65]. This database includes rich gesture variations, the samples in first class are shown in Fig.9. We used first three images for training and the rest for test. Moreover, all images were downsampled to 50×40 . Table 4 shows all comparison results. The recognition rate of RSSR is the highest among these methods which reaches 91.1% and is the only method whose recognition rate is over 90%. The other supervised spares method, TPTSR, has the second highest recognition accuracy 89.6%. All robust regression methods, RRC1, RRC2, ProCRC, NMR, and CESR, show similar performance.

Fig. 9: Sample images of the first subject in ORL.

Discussions. It can be concluded that the performance of some methods is not consistent. For instance, LRC is adept at dealing with illumination variations, but it performs poorly with expression changes and posture changes; the recognition rate of RRC1 is the highest on AR database, but falls significantly if the facial images include illumination changes and posture variations. In contrast, RSSR shows good generalization performance that it always shows top recognition accuracy among the compared eight methods (the respective rankings are 2nd, 3rd, 1st, and 1st in the four experiments). In practical, FR applications, the merit of RSSR is very important because the variations on facial image are

Table 4: Results for the ORL database

Method	CESR	RRC1	RRC2	LRC	SRC	CRC	TPTSR	ProCRC	NMR	**RSSR**
Recognition rate(%)	87.1	87.5	85.4	83.2	86.8	83.9	89.6	88.2	88.6	**91.1**

very hard to predict, which requires consistent good performance.

5.3. Face recognition with corruption and occlusion

It is well known that face pictures are susceptible to two kinds of noise: occlusions and corruptions. In this case, face recognition becomes a more challenging problem due to the occlusion and corruption are very varied. In the experiments, we verified the robustness of RSSR against to two typical facial image contaminations, namely random pixel corruption and random block occlusion.

FR with pixel corruption. As in [35], we evaluated the performance of RSSR in FR with random pixel corruption on Extended Yale B. To implement a more challenging experiment, we used less training samples, i.e. only samples from subset 1, and the test samples were from subset 3. We resized the images to 50×40. On each test image, a certain percentage of randomly selected pixels were replaced by corruption pixels whose values were uniformly chosen from 0 or twice as the biggest pixel value of the test image.

A representative example of RSSR with random corruption is presented in Fig.10.The test image Fig.10(b) was produced by adding random corrupted 60% pixels on the original facial shown in Fig.10(a). In this case, Fig.10(b) is very hard to recognize by human eyes. The prediction of RSSR is shown in Fig.10(e) which is rather satisfying. We can observe that not only these random noises on this image were eliminated, but also the illumination variations were mitigated. Fig.10(c) is a match residual image between the test facial image and the reconstructed image, which indicates that corruptions were separated effectively by RSSR. The coding coefficients (Fig.10(d)) shown that coefficients belonging to the correct subject had very large contribution in the representation while the coefficient vector was quite sparse.

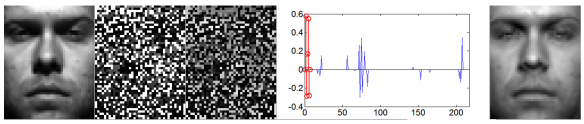


Fig. 10: Recognition under 60% random corruptions.(a) Original image for test from subset 3 on Extended Yale B database.(b) The test image with corruption.(c) Estimated error image.(d) Estimated representation coefficients by RSSR.(e) Reconstructed prediction images of RSSR.

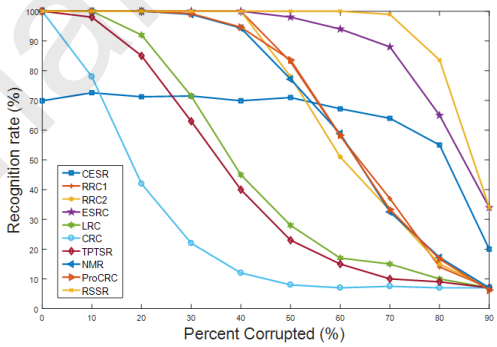


Fig. 11: The comparison of robustness of several algorithms against random pixel corruptions.

In Fig.11, the recognition rates of CESR, RRC1, RRC2, ESRC (Extended SRC), LRC, CRC, TPTSP, ProCRC, NMR and RSSR versus different percentages of corrupted pixels are plotted. The performance of CESR under illumination conditions is the lowest when the corruption is not serious, but it drops very late, which shows its good robustness but poor accuracy. RRC1 and RRC2 show some robustness which obtain 100% accuracy before 50% corruption level, but their performance degenerate severely after that. The ESRC, which incorporated an extended matrix into the original SRC to rectify the severe noises, kept high accuracy until corruption level exceeded 60%. For our RSSR, the curve of recognition rate keeps straight at 100% correct rate and only starts to bend until 70% pixels are corrupted which indicates the best robustness of RSSR among all the compared methods. In addition, the performance of LRC, CRC, and TPTSR sharply drops when corrupted pixels exceed 10%.

FR with block occlusion. In this section, the performance of RSSR dealing with random block occlusion was evaluated on Extended Yale B. Fig.12 presents an example (the first subject) to demonstrate the robustness of RSSR in this case. Fig.12(a) is a sample image and Fig.12(b) is its corresponding occluded counterpart for test. We used RSSR regression to produce the prediction of Fig.12(b). We can see that the block noise on the test image is eliminated by RSSR quite well, which is simultaneously proved by the error image Fig.12(c) in which the block occlusion is detected and highlighted clearly. From Fig.12(d), the correct class contributed mainly to represent the occluded face image which insures the correctness of the final decision.

To compare RSSR with other methods, we plot the recognition rate of CESR, RRC1, RRC2, ESRC, LRC, CRC, TPTSP, ProCRC, NMR, and RSSR under size of the occlusion from 0% to 50% in Fig.13. Except CESR, these methods acquire 100% recognition rate without block occlusion. With increase of size of the occlusion, the recognition rates of LRC, CRC, and TPTSR go down sharply, while the recognition rates of RRC1, RRC2, ESRC, and RSSR maintain 100% up to 20% occlusion. From 30% to 50% occlusion, the recognition rates of RSSR,

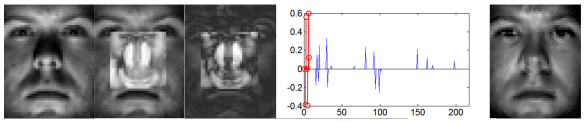


Fig. 12: Recognition of RSSR under 30% block occlusion.(a) Original image from subset 1 on Extended Yale B database. (b) Test image with occlusion. (c) Estimated error image. (d) Estimated representation coefficients of RSSR. (e) Reconstructed images by RSSR.

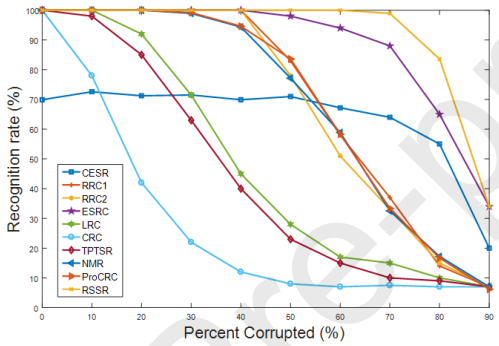


Fig. 13: Recognition rates of these methods under different levels of block occlusion

RRC1, and RRC2 preform more stable than ESRC.

6. Conclusions

A novel robust linear representation based model, RSSR, is proposed for robust face recognition in this paper. RSSR uses a two-phase collaborative representation to implement supervised sparse representation. As to tolerate possible variations on probe images, we use Huber loss as fidelity term in cost function for linear representation, which is capable of reserving more samples of correct class into the candidate set for the latter representation. Then the second phase of representation highlights the contribution of the correct class representing the probe, which is another key point that underlies the high performance of RSSR. Moreover, to solve the RSSR regression function we introduce two variables which can simplify the object function of RSSR in the process of optimiza-

tion. As shown in experiments, we compare RSSR classification method with the other state-of-the-art methods (e.g. SRC, LRC, TPTSR, CESR, RRC1, RRC2, ProCRC, NMR) under different conditions, including illumination variations, gesture changes, expression changes, corruptions, and occlusions. The performance of RSSR always ranks in the forefront of different comparisons and especially RSSR surpasses TPTSR, the original supervised sparse method, in all cases.

Acknowledgments

This work is sponsored by Natural Science Foundation of Chongqing, China (under Grant Nos. cstc2018jcyjAX0532 and cstc2014jcyjA40011).

References

[1] L. Shang, D.-S. Huang, J.-X. Du, C.-H. Zheng, Palmprint recognition using fastica algorithm and radial basis probabilistic neural network, Neurocomputing 69 (13-15) (2006) 1782–1786.

[2] D.-S. Huang, J.-X. Du, A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks, IEEE Transactions on neural networks 19 (12) (2008) 2099–2115.

[3] A. Anjos, M. M. Chakka, S. Marcel, Motion-based counter-measures to photo attacks in face recognition, Iet Biometrics 3 (3) (2014) 147–158.

[4] B. Li, C. H. Zheng, D. S. Huang, Locally linear discriminant embedding: An efficient method for face recognition, Pattern Recognition 41 (12) (2008) 3813–3821.

[5] Z. Q. Zhao, D. S. Huang, B. Y. Sun, Human face recognition based on multi-features using neural networks committee, Pattern Recognition Letters 25 (12) (2004) 1351–1358.

[6] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering.

[7] L.-F. Zhou, Y.-W. Du, W.-S. Li, J.-X. Mi, X. Luan, Pose-robust face recognition with huffman-lbp enhanced by divide-and-rule strategy, Pattern Recognition 78 (2018) 43–55.

[8] J. Yang, D. Zhang, A. F. Frangi, J. Y. Yang, Two-dimensional pca: a new approach to appearance-based face representation and recognition., IEEE Trans Pattern Anal Mach Intell 26 (1) (2004) 131–137.

[9] Z. Lai, Y. Xu, J. Yang, L. Shen, D. Zhang, Rotational invariant dimensionality reduction algorithms., IEEE Transactions on Cybernetics 47 (11) (2017) 3733–3746.

[10] Y. Lu, Z. Lai, Y. Xu, X. Li, D. Zhang, C. Yuan, Low-rank preserving projections., IEEE Trans Cybern 46 (8) (2016) 1900–1913.

[11] Y. Xu, D. Zhang, J. Yang, J. Y. Yang, An approach for directly extracting features from matrix data and its application in face recognition, Neurocomputing 71 (10) (2008) 1857–1865.

[12] Z. Lai, Y. Xu, Q. Chen, J. Yang, D. Zhang, Multilinear sparse principal component analysis, IEEE Trans Neural Netw Learn Syst 25 (10) (2014) 1942–1950.

[13] Z. Lai, W. K. Wong, Y. Xu, J. Yang, D. Zhang, Approximate orthogonal sparse embedding for dimensionality reduction., IEEE Transactions on Neural Networks and Learning Systems 27 (4) (2016) 723.

[14] Z. Zhang, Y. Xu, L. Shao, J. Yang, Discriminative block-diagonal representation learning for image recognition, IEEE transactions on neural networks and learning systems 29 (7) (2018) 3111–3125.

[15] X. F. Wang, D. S. Huang, H. Xu, An efficient local chanvese model for image segmentation, Pattern Recognition 43 (3) (2010) 603–618.

[16] X.-F. Wang, D.-S. Huang, A novel density-based clustering framework by using level set method, IEEE Transactions on knowledge and data engineering 21 (11) (2009) 1515–1531.

[17] D.-s. Huang, Radial basis probabilistic neural networks: Model and application, International Journal of Pattern Recognition and Artificial Intelligence 13 (07) (1999) 1083–1101.

[18] D.-S. Huang, H. H. Ip, Z. Chi, A neural root finder of polynomials based on root moments, Neural Computation 16 (8) (2004) 1721–1762.

[19] D.-S. Huang, Z. Chi, W.-C. Siu, A case study for constrained learning neural root finders, Applied mathematics and computation 165 (3) (2005) 699–718.

[20] D.-S. Huang, The local minima-free condition of feedforward neural networks for outer-supervised learning, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 28 (3) (1998) 477–480.

[21] D.-S. Huang, The united adaptive learning algorithm for the link weights and shape parameter in rbfn for pattern recognition, International journal of pattern recognition and artificial intelligence 11 (06) (1997) 873–888.

[22] M. Kirby, L. Sirovich, Application of the karhunen-loeve procedure for the characterization of human faces, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1) (2002) 103–108.

[23] S. W. Park, M. Savvides, A multifactor extension of linear discriminant analysis for face recognition under varying pose and illumination, Eurasip Journal on Advances in Signal Processing 2010 (1) (2010) 1–11.

[24] Comon, Pierre, Independent component analysis, a new concept?, Signal Processing 36 (3) (1994) 287–314.

[25] Z. Zhang, L. Liu, F. Shen, H. T. Shen, L. Shao, Binary multi-view clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (7) (2019) 1774–1782.

[26] Z. Li, Z. Zhang, J. Qin, Z. Zhang, L. Shao, Discriminative fisher embedding dictionary learning algorithm for object recognition, IEEE Transactions on Neural Networks and Learning Systems (2019) 1–15.

[27] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: Which helps face recognition?, in: International Conference on Computer Vision, 2012, pp. 471–478.

[28] Z. Zhang, Y. Xu, J. Yang, X. Li, D. Zhang, A survey of sparse representation: Algorithms and applications, IEEE Access 3 (2017) 490–530.

[29] W. Jie, X. Fang, J. Cui, L. Fei, Y. Ke, C. Yan, X. Yong, Robust sparse linear discriminant analysis, IEEE Transactions on Circuits & Systems for Video Technology PP (99) (2018) 1–1.

[30] Z. Zhang, L. Shao, Y. Xu, L. Liu, J. Yang, Marginal representation learning with graph structure self-adaptation, IEEE Transactions on Neural Networks & Learning Systems PP (99) (2017) 1–15.

[31] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (11) (2010) 2106–2112.

[32] I. Naseem, R. Togneri, M. Bennamoun, Robust regression for face recognition, in: International Conference on Pattern Recognition, 2012, pp. 1156–1159.

[33] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 210–227.

[34] R. He, W. S. Zheng, B. G. Hu, Maximum correntropy criterion for robust face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (8) (2011) 1561–1576.

[35] M. Yang, T. Song, F. Liu, L. Shen, Structured regularized robust coding for face recognition, IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society 22 (5) (2013) 1753–1766.

[36] Y. Xu, D. Zhang, J. Yang, J. Y. Yang, A two-phase test sample sparse representation method for use with face recognition, IEEE Transactions on Circuits and Systems for Video Technology 21 (9) (2011) 1255–1262.

[37] Y. Xu, W. Zuo, Z. Fan, Supervised sparse representation method with a heuristic strategy and face recognition experiments, Elsevier Science Publishers B. V., 2012.

[38] Y. Xu, Q. Zhu, Z. Fan, D. Zhang, J. Mi, Z. Lai, Using the idea of the sparse representation to perform coarse-to-fine face recognition, Information Sciences 238 (7) (2013) 138–148.

[39] X. Shi, Y. Yang, Z. Guo, Z. Lai, Face recognition by sparse discriminant analysis via joint l2,1 -norm minimization, Pattern Recognition 47 (7) (2014) 2447–2453.

[40] J.-X. Mi, Q. Fu, W. Li, Adaptive class preserving representation for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7427–7435.

[41] T. Liu, J.-X. Mi, Y. Liu, C. Li, Robust face recognition via sparse boosting representation, Neurocomputing 214 (2016) 944–957.

[42] R. He, W. Zheng, B. Hu, X. Kong, A regularized correntropy framework for robust pattern recognition, Neural Computation 23 (8) (2011) 2074–2100.

[43] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, Y. Ma, Toward a practical face recognition system: Robust alignment and illumination by sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2) (2012) 372–386.

[44] J. Qian, J. Yang, Y. Xu, General regression and representation model for classification., Plos One 9 (12) (2014) 166–172.

[45] J. Wang, C. Lu, M. Wang, P. Li, S. Yan, X. Hu, Robust face recognition via adaptive sparse representation., IEEE Transactions on Cybernetics 44 (12) (2014) 2368–2378.

[46] D. L. Donoho, For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution, Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences 59 (6) (2006) 797–829.

[47] E. J. Candes, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, IEEE Transactions on Information Theory 52 (12) (2006) 5406–5425.

[48] E. J. Cands, J. K. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Communications on Pure and Applied Mathematics 59 (8) (2006) 1207–1223.

[49] R. Tibshirani, Regression shrinkage and selection via the lasso: a retrospective, Journal of the Royal Statistical Society 73 (3) (2011) 273–282.

[50] P. Zhao, B. Yu, On model selection consistency of lasso, Journal of Machine Learning Research 7 (12) (2006) 2541–2563.

[51] P. J. Huber, Robust regression: Asymptotics, conjectures and monte carlo, Annals of Statistics 1 (5) (1973) 799–821.

[52] Z. Zhang, Parameter estimation techniques: a tutorial with application to conic fitting, Water Research 39 (15) (1997) 3686–3696.

[53] W. Dong, H. Lu, M. H. Yang, Least soft-threshold squares tracking, in: IEEE Conference on Computer Vision & Pattern Recognition, 2013.

[54] K. Madsen, H. B. Nielsen, Finite alogorithms for robust linear regression, BIT Computer Science and Numerical Mathematics, 1990.

[55] Shewchuk, R. Jonathan, An introduction to the conjugate gradient method without the agonizing pain, Technical Report CMU-CS-94-125 186 (3) (1994) 219–20.

[56] S. Cai, Z. Lei, W. Zuo, X. Feng, A probabilistic collaborative representation based approach for pattern classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[57] J. Yang, L. Luo, J. Qian, Y. Tai, F. Zhang, Y. Xu, Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (1) (2016) 156–171.

[58] R. He, W. S. Zheng, T. Tan, Z. Sun, Half-quadratic based iterative minimization for robust sparse representation., IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (2) (2014) 261–75.

[59] M. Nikolova, M. K. Ng, Analysis of Half-Quadratic Minimization Methods for Signal and Image Recovery, Society for Industrial and Applied Mathematics, 2005.

[60] J. X. Mi, Face image recognition via collaborative representation on selected training samples, Optik - International Journal for Light and Electron Optics 124 (18) (2013) 3310–3313.

[61] A. M. Martinez, The ar face database, Cvc Technical Report 24.

[62] K. C. Lee, J. Ho, D. J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Trans Pattern Anal Mach Intell 27 (5) (2005) 684–698.

[63] A. S. Georghiades, P. N. Belhumeur, D. J. Kriegman, From few to many: Illumination cone models for face recognition under variable lighting and pose, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 643–660.

[64] P. J. Phillips, H. Wechsler, J. Huang, P. J. Rauss, The feret database and evaluation procedure for face-recognition algorithms, Image and Vision Computing J 16 (5) (1998) 295–306.

[65] F. S. Samaria, A. C. Harter, Parameterisation of a stochastic model for human face identification, in: Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on, 1994, pp. 138–142.