Sparse Representation or Collaborative Representation: Which Helps Face Recognition?

Lei Zhang[a], Meng Yang[a], and Xiangchu Feng[b]
[a]Dept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China
[b]Dept. of Applied Mathematics, Xidian University, Xi'an, China
{cslzhang, csmyang}@comp.polyu.edu.hk

Abstract

As a recently proposed technique, sparse representation based classification (SRC) has been widely used for face recognition (FR). SRC first codes a testing sample as a sparse linear combination of all the training samples, and then classifies the testing sample by evaluating which class leads to the minimum representation error. While the importance of sparsity is much emphasized in SRC and many related works, the use of collaborative representation (CR) in SRC is ignored by most literature. However, is it really the l_1-norm sparsity that improves the FR accuracy? This paper devotes to analyze the working mechanism of SRC, and indicates that it is the CR but not the l_1-norm sparsity that makes SRC powerful for face classification. Consequently, we propose a very simple yet much more efficient face classification scheme, namely CR based classification with regularized least square (CRC_RLS). The extensive experiments clearly show that CRC_RLS has very competitive classification results, while it has significantly less complexity than SRC.

1. Introduction

It has been found that natural images can be sparsely coded by structural primitives [1], and in recent years sparse coding or sparse representation has been widely studied to solve the inverse problems in various image restoration applications [2-3], partially due to the progress of l_0-norm and l_1-norm minimization techniques [4-6].

Recently, sparse representation has also been used in pattern classification. Huang *et al.* [7] sparsely coded a signal over a set of redundant bases and classified the signal based on its coding vector. In [8], Wright *et al.* reported a very interesting work by using sparse representation for robust face recognition (FR). A query face image is first sparsely coded over the template images, and then the classification is performed by checking which class yields the least coding error. Such a sparse representation based classification (SRC) scheme achieves a great success in FR, and it boosts the research of sparsity based pattern classification. Gao *et al.* [9] proposed the kernel sparse representation for FR, while Yang and Zhang [10] used the Gabor features for SRC with a learned Gabor occlusion dictionary to reduce the computational cost. Cheng *et al.* [11] discussed the l_1-graph for classification, and Yang *et al.* [12] combined sparse coding with linear spatial pyramid matching for image classification. A recent review of sparse representation for computer vision and pattern recognition applications can be found in [13].

In sparse representation based FR, usually we assume that the face images are aligned. Recently, sparse representation has been extended to solve the misalignment or pose change. The method in [14] is invariant to image-plane transformation. The method in [15] could deal with misalignment and illumination variation. In [16], Peng *et al.* studied how to simultaneously align a batch of linearly correlated images with gross corruption.

Sparse representation (or coding) codes a signal $\boldsymbol{y}$ over a dictionary $\boldsymbol{\Phi}$ such that $\boldsymbol{y} \approx \boldsymbol{\Phi\alpha}$ and $\boldsymbol{\alpha}$ is a sparse vector. The sparsity of $\boldsymbol{\alpha}$ can be measured by l_0-norm, which counts the number of non-zeros in $\boldsymbol{\alpha}$. Since the combinatorial l_0-minimization is NP-hard, the l_1-minimization, as the closest convex function to l_0-minimization, is widely employed in sparse coding: $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\|\boldsymbol{y} - \boldsymbol{\Phi\alpha}\|_2 \le \varepsilon$, where ε is a small constant. Although l_1-minimization is much more efficient than l_0-minimization, it is still time consuming, and hence many fast algorithms were proposed to speed up the l_1-minimization process.

As reviewed in [17], there are five representative fast l_1-minimization approaches: Gradient Projection, Homotopy, Iterative Shrinkage-Thresholding, Proximal Gradient, and Augmented Lagrange Multiplier (ALM). It was indicated that for noisy data, first order l_1-minimization techniques (e.g., SpaRSA [18], FISTA [19] and ALM [20]) are more efficient, while for FR, Homotopy [21], ALM and l_1_*ls* [22] are better for their accuracy and fast speed.

Although SRC [8] has shown interesting results in FR and has been widely studied in the community, its working mechanism has not been clearly revealed yet. Most literature, including [8], emphasizes too much on the role of l_1-norm sparsity in face classification, while the role of collaborative representation (CR), i.e., using the training samples from all classes to represent the query sample $\boldsymbol{y}$, is much ignored. The l_1-minimization makes the sparsity

based classification schemes such as SRC very expensive; however, is it really the l_1-norm sparsity that makes SRC powerful for FR? Very recently some researchers have started to question the use of sparsity in image classification, such as [29-30].

This paper devotes to analyze the working mechanism of SRC. We will explain why sparsity could improve discrimination, and more importantly, we will indicate that it is the CR, but not the l_1-norm sparsity, that plays the essential role for classification in SRC. Consequently, we propose a new classification scheme, namely CR based classification with regularized least square (CRC_RLS), which has significantly less complexity than SRC but leads to very competitive classification results.

Section 2 briefly reviews SRC. Section 3 analyzes sparse representation and CR. Section 4 presents the CRC_RLS scheme. Section 5 conducts extensive experiments, and Section 6 concludes the paper.

2. The SRC scheme

Table 1: The SRC Algorithm

1. Normalize the columns of $\boldsymbol{X}$ to have unit l_2-norm.
2. Code $\boldsymbol{y}$ over $\boldsymbol{X}$ via l_1-minimization
$$(\hat{\boldsymbol{\alpha}}) = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \text{ s.t. } \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_2 < \varepsilon \quad (1)$$
where constant ε is to account for the dense small noise in $\boldsymbol{y}$, or to balance the coding error of $\boldsymbol{y}$ and the sparsity of $\boldsymbol{\alpha}$.
3. Compute the residuals
$$e_i(\boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{X}_i\hat{\boldsymbol{\alpha}}_i\|_2 \quad (2)$$
where $\hat{\boldsymbol{\alpha}}_i$ is the coding coefficient vector associated with class i.
4. Output the identity of $\boldsymbol{y}$ as
$$\text{identity}(\boldsymbol{y}) = \arg\min_i \{e_i\} \quad (3)$$

Denote by $\boldsymbol{X}_i \in \Re^{m\times n}$ the dataset of the i^{th} class, and each column of $\boldsymbol{X}_i$ is a sample of class i. Suppose that we have K classes of subjects, and let $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K]$. Once a query image $\boldsymbol{y}\in\Re^m$ comes, we code it as $\boldsymbol{y}\approx\boldsymbol{X}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}=[\boldsymbol{\alpha}_1;\ldots,\boldsymbol{\alpha}_i;\ldots;\boldsymbol{\alpha}_K]$ and $\boldsymbol{\alpha}_i$ is the coding vector associated with class i. If $\boldsymbol{y}$ is from the i^{th} class, usually $\boldsymbol{y}\approx\boldsymbol{X}_i\boldsymbol{\alpha}_i$ holds well, implying that most coefficients in $\boldsymbol{\alpha}_k$, $k\neq i$, are nearly zeros and only $\boldsymbol{\alpha}_i$ has significant entries. That is, the sparse non-zero entries in $\boldsymbol{\alpha}$ can encode the identity of sample $\boldsymbol{y}$. The procedures of SRC are summarized in Table 1.

3. Sparse representation and collaborative representation

From Table 1, we see that there are two key points in SRC. The first key point is that the coding vector of query sample $\boldsymbol{y}$ is required to be sparse, and the second key point is that the coding of $\boldsymbol{y}$ is performed collaboratively over the whole dataset $\boldsymbol{X}$ instead of each subset $\boldsymbol{X}_i$. Suppose that $\boldsymbol{y}$ belongs to some class in the dataset, it was claimed in [8] that the sparsest (or the most compact) representation of $\boldsymbol{y}$ over $\boldsymbol{X}$ is naturally discriminative and thus can indicate the identity of $\boldsymbol{y}$. It was also claimed that SRC is a generalization of the classical nearest neighbor (NN) and nearest subspace (NS) classifiers. The NN classifier represents $\boldsymbol{y}$ by each individual of the training samples; the NS classifier represents $\boldsymbol{y}$ by the training samples of each class; and SRC represents $\boldsymbol{y}$ collaboratively by samples of all classes. In this section, we first illustrate why sparsity makes representation more discriminative, and then discuss the collaborative representation involved in SRC.

3.1. Why sparse representation?

Denote by $\boldsymbol{\Phi}\in\Re^{m\times n}$ a dictionary of atoms. If $\boldsymbol{\Phi}$ is complete, then any signal $\boldsymbol{x}\in\Re^m$ can be accurately represented as the linear combination of the atoms in $\boldsymbol{\Phi}$. If $\boldsymbol{\Phi}$ is orthogonal, however, often we need to use many atoms from $\boldsymbol{\Phi}$ to faithfully represent $\boldsymbol{x}$. If we want to use less atoms to represent $\boldsymbol{x}$, we must relax the orthogonality imposed on $\boldsymbol{\Phi}$. In other words, we must allow more atoms to be involved in $\boldsymbol{\Phi}$ so that we have more choices to represent $\boldsymbol{x}$, leading to an over-complete dictionary $\boldsymbol{\Phi}$ but a sparser representation of signal $\boldsymbol{x}$. For example, it is well-known that redundant wavelet transforms have much better denoising performance than orthogonal wavelet transforms. The great success of sparse representation in image restoration [2-3] further validates this.

In the scenario of FR, each class of face images often lies in a small subspace of $\Re^m$. That is, the m-dimensional face image $\boldsymbol{x}$ can be characterized by a feature vector of much lower dimensionality. If we take the set of training samples of class i, i.e., $\boldsymbol{X}_i$, as the dictionary for this class, in practice the atoms (i.e., the training samples) of $\boldsymbol{X}_i$ will be correlated. Assume that we have enough training samples for each class so that all the images of class i can be faithfully represented by $\boldsymbol{X}_i$, then $\boldsymbol{X}_i$ is an over-complete dictionary[1] because of the correlation of training samples of class i, and we can conclude that a testing sample $\boldsymbol{y}$ of class i can be sparsely represented over dictionary $\boldsymbol{X}_i$.

Another important fact in FR is that all the face images are somewhat similar, while some subjects may have very similar face images. This implies that dictionary $\boldsymbol{X}_i$ and dictionary $\boldsymbol{X}_j$ are not incoherent but can be highly correlated. Let $\boldsymbol{X}_j = \boldsymbol{X}_i + \boldsymbol{\Delta}$. Using the NS classifier, for a query sample $\boldsymbol{y}$ from class i, we can calculate by least square method a vector $\boldsymbol{\alpha}_i = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{X}_i\boldsymbol{\alpha}\|_2$. Let $\boldsymbol{e}_i =$

[1] More strictly speaking, it should be the dimensionality reduced dictionary of $\boldsymbol{X}_i$ that is over-complete. For the convenience of expression, we simply use $\boldsymbol{X}_i$ in the development.

$\boldsymbol{y}$-$\boldsymbol{X}_i\boldsymbol{\alpha}_i$. Similarly, if we represent $\boldsymbol{y}$ by class j, there is $\boldsymbol{\alpha}_j = \arg\min_{\boldsymbol{\alpha}} \left\|\boldsymbol{y} - \boldsymbol{X}_j\boldsymbol{\alpha}\right\|_2$ and we let $\boldsymbol{e}_j$=$\boldsymbol{y}$-$\boldsymbol{X}_j\boldsymbol{\alpha}_j$. Suppose that $\boldsymbol{X}_i, \boldsymbol{X}_j \in \Re^{m\times n}$, if $\boldsymbol{\Delta}$ is small such that

$$\xi = \frac{\left\|\boldsymbol{\Delta}\right\|_F}{\left\|\boldsymbol{X}_i\right\|_F} \le \frac{\sigma_n(\boldsymbol{X}_i)}{\sigma_1(\boldsymbol{X}_i)}$$

where $\sigma_1(\boldsymbol{X}_i)$ and $\sigma_n(\boldsymbol{X}_i)$ are the largest and smallest eigenvalues of $\boldsymbol{X}_i$, respectively, then we have the following relationship between $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$ (page 242, [28]):

$$\frac{\left\|\boldsymbol{e}_j - \boldsymbol{e}_i\right\|_2}{\left\|\boldsymbol{y}\right\|_2} \le \xi\left(1+\kappa_2(\boldsymbol{X}_i)\right)\min\left\{1, m-n\right\} + O\left(\xi^2\right) \quad (4)$$

where $\kappa_2(\boldsymbol{X}_i)$ is the l_2-norm conditional number of $\boldsymbol{X}_i$.

From Eq. (4), we can clearly see that if $\boldsymbol{\Delta}$ is small, i.e., subjects i and j look similar to each other, then the distance between $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$ can be very small. This makes the classification very unstable because a small disturbance can lead to $\|\boldsymbol{e}_j\|_2$<$\|\boldsymbol{e}_i\|_2$, resulting in a wrong classification.

The above problem can be much alleviated by imposing some sparsity on $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_j$. The reason is very simple. If $\boldsymbol{y}$ is from class i, it is more likely that we can use only a few samples (e.g., 5 or 6 samples) in $\boldsymbol{X}_i$ to represent $\boldsymbol{y}$ with a good accuracy. In contrast, we may need more samples (e.g., 8 or 9 samples) in $\boldsymbol{X}_j$ to represent $\boldsymbol{y}$ with nearly the same accuracy. Under a certain sparsity constraint, the representation error of $\boldsymbol{y}$ by $\boldsymbol{X}_i$ will be visibly lower than that by $\boldsymbol{X}_j$, making the classification of $\boldsymbol{y}$ easier. The sparse representation of $\boldsymbol{y}$ by $\boldsymbol{\Phi}$ can be formulated as

$$\min_{\boldsymbol{\alpha}} \left\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\right\|_2 \quad \text{s.t.} \quad \left\|\boldsymbol{\alpha}\right\|_p \le \varepsilon \quad (5)$$

where ε is a constant and p can be 0, 1, or any other eligible sparsity metric.

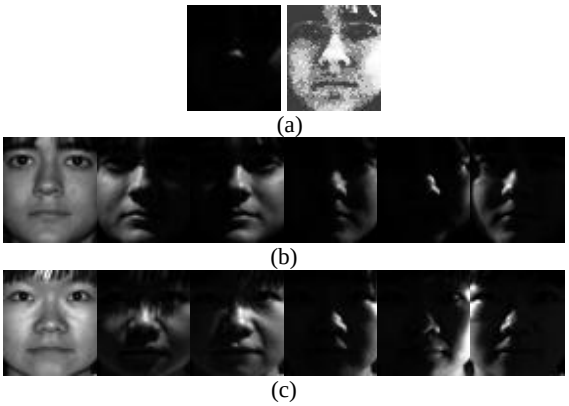

Figure 1: (a) The query face image (left: original image, right: the one after histogram equalization for better visualization); (b) some training samples from the class of the query image; (c) some training samples from another class.

Let $e = \left\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\right\|_2$ and set p=0 in Eq. (5). We could plot the curve of e versus ε to illustrate why sparsity improves discrimination. Fig. 1(a) shows a testing face image from class 32 in the Extended Yale B database. Some training samples of class 32 are shown in Fig. 1(b). Some training samples of class 5, which looks similar to class 32, are shown in Fig. 1(c). We use the training samples of the two classes as dictionaries to represent the query sample in Fig. 1(a), respectively, under different sparsity ε. The two "e vs. ε" curves are drawn in Fig. 2.

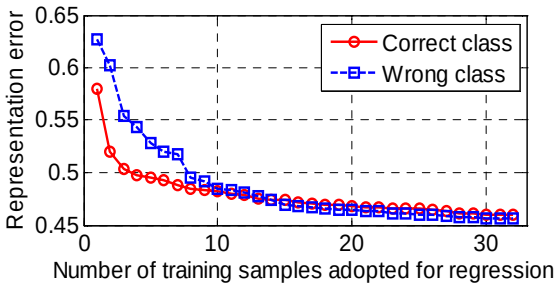

Figure 2: The curve of representation error versus the number of training samples in each class.

From Fig. 2, we can see that when using only a few training samples (<3) to represent the query sample, both of the two classes have big error; when more and more training samples are involved, the representation error decreases. However, the discrimination ability of the representation error will also reduce if too many samples (>10) are used. Thus, the sparsity of coefficients should be considered. From the above analyses, we may have the following proposition for l_0-norm sparsity: *a query sample should be classified to the class which could faithfully represent it using less number of samples*.

In practice, if the number of training samples of each class is relatively big, we can represent the testing sample $\boldsymbol{y}$ class by class. Since l_0-minimization is combinatorial NP-hard, l_1-minimization with the following Lagrangian formulation is often adopted:

$$\left(\boldsymbol{\alpha}_i\right) = \arg\min_{\boldsymbol{\alpha}} \left\{\left\|\boldsymbol{y} - \boldsymbol{X}_i\boldsymbol{\alpha}\right\|_2^2 + \lambda\left\|\boldsymbol{\alpha}\right\|_1\right\} \quad (6)$$

Both the representation error e_i=$\|\boldsymbol{y}$-$\boldsymbol{X}_i\boldsymbol{\alpha}_i\|_2$ and the sparsity term $\|\boldsymbol{\alpha}_i\|_1$ can be used to classify the sample. Later we will see that (please refer to the experimental results in Section 5.2) actually the l_2-norm can also be used to regularize $\boldsymbol{\alpha}_i$, and the l_1-norm and l_2-norm lead to almost the same result.

3.2. Why collaborative representation?

In our discussion in Section 3.1, we assumed that there are enough training samples for each class so that the (dimensionality reduced) dictionary $\boldsymbol{X}_i$ is over-complete. Unfortunately, FR is a typical small-sample-size problem, and $\boldsymbol{X}_i$ is under-complete in general. If we use $\boldsymbol{X}_i$ to represent $\boldsymbol{y}$, the representation error can be big, even when

$\boldsymbol{y}$ is from class i. Consequently, the classification will be unstable not matter the error e_i or the sparsity $\|\boldsymbol{\alpha}_i\|_p$ or both of them are used for decision making.

One obvious solution to solving this problem is to use more samples of class i to represent $\boldsymbol{y}$. But where could we have these samples? Fortunately, one fact in FR is that face images of different classes share similarities. Some sample from class j may be very helpful to represent the testing sample with label i. In SRC [8], this "lack of samples" problem is solved by taking the face images from all the other classes as the possible samples of each class. That is, it codes the testing image $\boldsymbol{y}$ collaboratively over the dictionary of all samples $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K]$ under the l_1-norm sparsity constraint.

One interesting point here is that after the collaborative representation (CR) with all classes, SRC classifies $\boldsymbol{y}$ individually (i.e., check class by class). For the simplicity of analysis, let's remove the l_1-norm sparsity term in Eq. (1), and then the representation becomes a least square problem: $(\hat{\boldsymbol{\alpha}}) = \arg\min_{\alpha} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_2^2$. The associated representation $\hat{\boldsymbol{y}} = \sum_i \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i$ is actually the perpendicular projection of $\boldsymbol{y}$ onto the space spanned by $\boldsymbol{X}$. In SRC, the reconstruction error by each class $e_i = \|\boldsymbol{y} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i\|_2^2$ is used for classification. It can be readily derived that

$$e_i = \|\boldsymbol{y} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i\|_2^2 = \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2 + \|\hat{\boldsymbol{y}} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i\|_2^2$$

Obviously, it is the amount $e_i^* = \|\hat{\boldsymbol{y}} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}\|_2^2$ that works for classification because $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2$ is a constant for all classes.

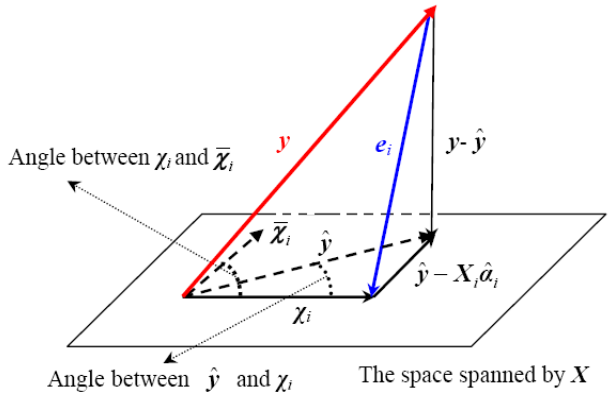

Figure 3: Geometric illustration of the representation of $\boldsymbol{y}$ over $\boldsymbol{X}$.

Denote by $\boldsymbol{\chi}_i = \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i$ and $\bar{\boldsymbol{\chi}}_i = \sum_{j \neq i} \boldsymbol{X}_j \hat{\boldsymbol{\alpha}}_j$. Fig. 3 shows geometrically the representation of $\boldsymbol{y}$ over $\boldsymbol{X}$. Since $\bar{\boldsymbol{\chi}}_i$ is parallel to $\hat{\boldsymbol{y}} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i$, we can readily have

$$\frac{\|\hat{\boldsymbol{y}}\|_2}{\sin(\boldsymbol{\chi}_i, \bar{\boldsymbol{\chi}}_i)} = \frac{\|\hat{\boldsymbol{y}} - \boldsymbol{X}_i \hat{\boldsymbol{\alpha}}_i\|_2}{\sin(\hat{\boldsymbol{y}}, \boldsymbol{\chi}_i)}$$

where $(\boldsymbol{\chi}_i, \bar{\boldsymbol{\chi}}_i)$ is the angle between $\boldsymbol{\chi}_i$ and $\bar{\boldsymbol{\chi}}_i$, and $(\hat{\boldsymbol{y}}, \boldsymbol{\chi}_i)$ is the angle between $\hat{\boldsymbol{y}}$ and $\boldsymbol{\chi}_i$. Finally, the representation error can be represented by

$$e_i^* = \frac{\sin^2(\hat{\boldsymbol{y}}, \boldsymbol{\chi}_i) \|\hat{\boldsymbol{y}}\|_2^2}{\sin^2(\boldsymbol{\chi}_i, \bar{\boldsymbol{\chi}}_i)} \tag{7}$$

Eq. (7) shows that by using CR, when we judge if $\boldsymbol{y}$ belongs to class i, we will not only consider if the angle between $\hat{\boldsymbol{y}}$ and $\boldsymbol{\chi}_i$ is small (i.e., if $\sin(\hat{\boldsymbol{y}}, \boldsymbol{\chi}_i)$ is small), we will also consider if the angle between $\boldsymbol{\chi}_i$ and $\bar{\boldsymbol{\chi}}_i$ is big (i.e., if $\sin(\boldsymbol{\chi}_i, \bar{\boldsymbol{\chi}}_i)$ is big). Such a "double checking" makes the classification more effective and robust.

One problem is that when the number of classes is too big, the least square solution $(\hat{\boldsymbol{\alpha}}) = \min_{\alpha} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_2^2$ will become unstable. In SRC, the l_1-norm sparsity constraint is imposed on $\boldsymbol{\alpha}$ to make the solution stable. However, it is not necessary to use the strong l_1-norm to this end. As we will see next, by using the much weaker l_2-norm to regularize the solution of $\boldsymbol{\alpha}$, we can have similar classification results but with significantly lower complexity. *In summary, it is the CR but not the l_1-norm sparsity constraint that truly improves the FR performance.*

4. Collaborative representation based classification (CRC)

Most of the previous works [8-13] emphasize the importance of sparsity for classification but do not investigate much the role of collaboration between classes in representing the query sample. Is it really the sparsity that improves the FR accuracy? Or is it the CR that truly helps FR? To answer this question, we propose here a simple CR based classification (CRC) scheme, and conduct experiments to give the answer in next section.

In order to collaboratively represent the query sample using $\boldsymbol{X}$ with low computational burden, we propose to use the regularized least square method. There is

$$(\hat{\boldsymbol{\rho}}) = \arg\min_{\rho} \left\{ \|\boldsymbol{y} - \boldsymbol{X} \cdot \boldsymbol{\rho}\|_2^2 + \lambda \|\boldsymbol{\rho}\|_2^2 \right\} \tag{8}$$

where λ is the regularization parameter. The role of the regularization term is twofold. First, it makes the least square solution stable, and second, it introduces a certain amount of "sparsity" to the solution $\hat{\boldsymbol{\rho}}$, yet this sparsity is much weaker than that by l_1-norm.

The solution of CR with regularized least square in Eq. (8) can be easily and analytically derived as

$$\hat{\boldsymbol{\rho}} = \left(\boldsymbol{X}^T \boldsymbol{X} + \lambda \cdot \boldsymbol{I}\right)^{-1} \boldsymbol{X}^T \boldsymbol{y} \tag{9}$$

Let $\boldsymbol{P} = \left(\boldsymbol{X}^T \boldsymbol{X} + \lambda \cdot \boldsymbol{I}\right)^{-1} \boldsymbol{X}^T$. Clearly, $\boldsymbol{P}$ is independent of $\boldsymbol{y}$ so that it can be pre-calculated as a projection matrix. Once a query sample $\boldsymbol{y}$ comes, we can just simply project $\boldsymbol{y}$ onto $\boldsymbol{P}$

via $\boldsymbol{Py}$. This makes CR very fast.

The classification by $\hat{\boldsymbol{\rho}}$ is similar to the classification by $\hat{\boldsymbol{\alpha}}$ in SRC (refer to Table 1). In addition to the class specific representation residual $\|\boldsymbol{y}-\boldsymbol{X}_i\cdot\hat{\boldsymbol{\rho}}_i\|_2$, where $\hat{\boldsymbol{\rho}}_i$ is the coefficient vector associated with class i, the l_2-norm "sparsity" $\|\hat{\boldsymbol{\rho}}_i\|_2$ can also bring some discrimination information for classification. Therefore we propose to use both of them in classification. Based on our experiments, this improves slightly the classification accuracy over that by using only the representation residual. The proposed CRC with regularized least square (CRC_RLS) algorithm is summarized as follows.

Table 2: The CRC_RLS Algorithm

1. Normalize the columns of $\boldsymbol{X}$ to have unit l_2-norm.
2. Code $\boldsymbol{y}$ over $\boldsymbol{X}$ by
$$\hat{\boldsymbol{\rho}} = \boldsymbol{P}\boldsymbol{y}$$
where $\boldsymbol{P} = \left(\boldsymbol{X}^T\boldsymbol{X}+\lambda\cdot\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T$.
3. Compute the regularized residuals
$$r_i = \|\boldsymbol{y}-\boldsymbol{X}_i\cdot\hat{\boldsymbol{\rho}}_i\|_2 / \|\hat{\boldsymbol{\rho}}_i\|_2 \qquad (10)$$
4. Output the identity of $\boldsymbol{y}$ as
Identity($\boldsymbol{y}$) = argmin$_i\{r_i\}$.

5. Experimental results

Considering the accuracy and efficiency, we chose l_1_ls [22] to solve the l_1-regularized minimization in SRC. All the experiments were carried out using MATLAB on a 3.16 GHz machine with 3.25GB RAM. In the experiments of gender classification in Section 5.2, the parameter λ in CRC_RLS is set as 0.08. In FR, considering that when more classes (and thus more samples) are used, the least square solution of CR will be more unstable and thus higher regularization is required, we set λ as 0.001**n*/700 in all FR experiments, where n is the number of training samples. The MATLAB code of CRC_RLS can be downloaded at http://www4.comp.polyu.edu.hk/~cslzhang/code.htm.

Three face databases, including Extended Yale B [23] [24], AR [25], and large-scale Multi-PIE [26], are used to test the performance of CRC_RLS and its competing methods, including SRC [8], SVM, LRC (linear regression classification) [27] and NN. Note that LRC is actually an NS based method.

5.1. The role of sparsity: l_1 or l_2?

In this section, we study the role of sparsity in FR. Two representative face databases, Extended Yale B [23][24] and AR [25], are used (the experimental settings are described in Section 5.3). We use Eigenfaces of dimensionality 300 as the input facial features, and use all the training samples as the dictionary.

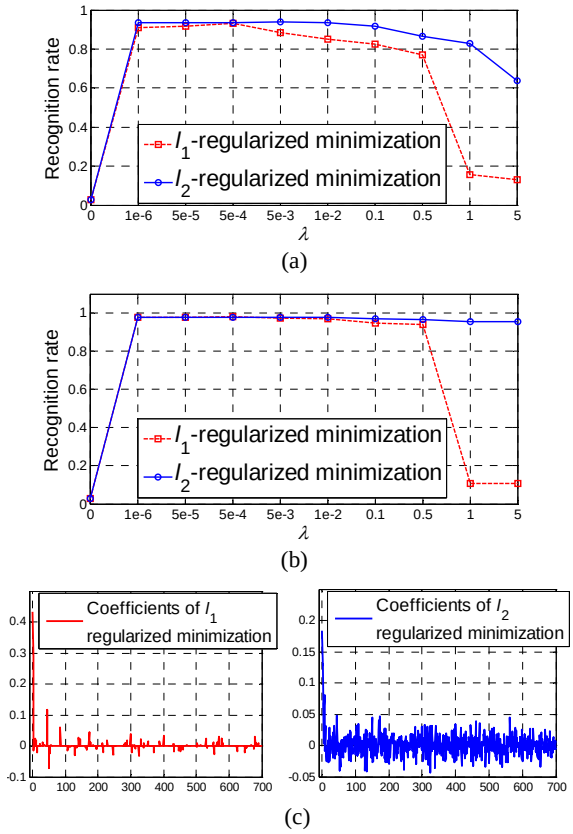

Figure 4: The recognition rates of SRC (l_1-regularized minimization) and CRC_RLS (l_2-regularized minimization) versus the different values of λ on the (a) AR and (b) Extended Yale B databases; (c) the coding coefficients of a query sample.

The sparse coding of SRC in Eq. (1) can be equivalently written as $(\hat{\boldsymbol{\alpha}}) = \arg\min_{\boldsymbol{\alpha}}\left\{\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\alpha}\|_2^2+\lambda\|\boldsymbol{\alpha}\|_1\right\}$. We test the performance of SRC (l_1-regulazied minimization) and CRC_RLS (l_2-regulazied minimization) by increasing the value of regularization parameter λ. The results on the AR and Extended Yale B databases are shown in Fig. 4(a) and Fig. 4(b), respectively. We can see that when λ=0, SRC and CRC_RLS fail. When λ is assigned a small positive value, e.g., from 0.000001 to 0.1, good results can be achieved by SRC and CRC_RLS. When λ is too big (e.g., >0.1) the recognition rates of both methods drop.

From Fig. 4 we can have the following findings. First, with the increase of sparsity (>0.000001), no much benefit on recognition rate can be gained. Second, l_2-regulazied minimization (i.e., CRC_RLS) could get higher recognition rates than l_1-regulazied minimization (i.e., SRC) in a broad range of λ. This implies that l_1-norm does not play the key role in face classification.

Fig. 4(c) plots the query sample's coding coefficients by SRC and CRC_RLS when they achieve their best results in the AR database. It can be seen that CRC_RLS has much

weaker sparsity than SRC; however, it achieves not worse results. Again, sparsity of the representation coefficients is useful but not that crucial for FR. *What really crucial is the CR mechanism in both CRC_RLS and SRC*.

5.2. Gender classification

In this section, we validate our claim in Section 3.1 that when the samples in each class are enough, there is no need to code the testing sample over the whole dictionary. We chose a non-occluded subset (14 images per subject) of AR [25] consisting of 50 male and 50 female subjects. Images of the first 25 males and 25 females were used for training, and the remaining images for testing. We used PCA to reduce the dimension of each image to 300. For this 2-class classification problem with enough training samples, we code the testing sample by each class' dictionary, and then classify it based on both the representation error and coefficient sparsity. That is, the query sample $\boldsymbol{y}$ is classified to the class which gives the minimal $r_i(\boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{X}_i\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$ or $r_i(\boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{X}_i\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_2^2$. The methods are then called L_1R (for l_1-regularized minimization) and L_2R (for l_2-regularized minimization).

We compare L_1R and L_2R with the CRC_RLS, SRC, SVM, LRC and NN, and the results are listed in Table 3. One can see that L_1R and L_2R get the best results, which validates that coding on each class' dictionary is more powerful than coding on the whole dictionary when the training samples of each class are enough, no matter l_1- or l_2-regularized minimization is used. CRC_RLS gets the second best result, about 1.4% higher than SRC.

Table 3: The results of different methods on gender classification using the AR database.

L_1R	**L_2R**	**CRC_RLS**	SRC	SVM	LRC	NN
94.9%	**94.9%**	**93.7%**	92.3%	92.4%	27.3%	90.7%

5.3. Face recognition

The proposed CRC_RLS is then tested for FR. The Eigenface is used as the face feature.

a) Extended Yale B Database: The Extended Yale B [23] [24] database contains about 2,414 frontal face images of 38 individuals. We used the cropped and normalized face images of size 54×48, which were taken under varying illumination conditions. We randomly split the database into two halves. One half, which contains 32 images for each person, was used as the dictionary, and the other half was used for testing. Table 4 shows the recognition rates versus feature dimension by NN, LRC, SVM, SRC and CRC_RLS. It can be seen that CRC_RLS and SRC achieve very similar results in all dimensions (the difference of recognition rate is less than 0.5%). Since there are relatively enough (32 per class) training samples, all the methods have not bad recognition rates.

Table 4: The face recognition results of different methods on the Extended Yale B database.

Dim	84	150	300
NN	85.8%	90.0%	91.6%
LRC	94.5%	95.1%	95.9%
SVM	94.9%	96.4%	97.0%
SRC	**95.5%**	**96.8%**	**97.9%**
CRC_RLS	95.0%	96.3%	**97.9%**

2) AR database: As in [8], a subset (with only illumination and expression changes) that contains 50 male subjects and 50 female subjects was chosen from the AR dataset [25] in our experiments. For each subject, the seven images from Session 1 were used for training, with the other seven images from Session 2 for testing. The images were cropped to 60×43. The comparison of competing methods is given in Table 5. We can see that CRC_RLS achieves the best result when the dimensionality is 120 or 300. The recognition rates of CRC_RLS and SRC are both at least 10% higher than other methods. This shows that CR does have much contribution to face classification.

Table 5: The face recognition results of different methods on the AR database.

Dim	54	120	300
NN	68.0%	70.1%	71.3%
LRC	71.0%	75.4%	76.0%
SVM	69.4%	74.5%	75.4%
SRC	**83.3%**	89.5%	93.3%
CRC_RLS	80.5%	**90.0%**	**93.7%**

Table 6: The face recognition results of different methods on the MPIE database.

	NN	LRC	SVM	SRC	CRC_RLS
S2	86.4%	87.1%	85.2%	93.9%	**94.1%**
S3	78.8%	81.9%	78.1%	**90.0%**	89.3%
S4	82.3%	84.3%	82.1%	**94.0%**	93.3%

3) Multi PIE database: The CMU Multi-PIE database [26] contains images of 337 subjects captured in four sessions with simultaneous variations in pose, expression, and illumination. In the experiments, all the 249 subjects in Session 1 were used. For the training set, we used the 14 frontal images with 14 illuminations [2] and neutral expression. For the testing sets, 10 typical frontal images[3] of illuminations taken with neutral expressions from Session 2 to Session 4 were used. The dimensionality of Eigenface is 300. Table 6 lists the recognition rates in three tests by the competing methods. The results validate that CRC_RLS and SRC are the best in accuracy, with at least

[2] Illuminations {0,1,3,4,6,7,8,11,13,14,16,17,18,19}.
[3] Illuminations {0,2,4,6,8,10,12,14,16,18}.

6% improvement than the other three methods.

4) FR with real face disguise: As in [8], a subset from the AR database consisting of 1400 images from 100 subjects, 50 male and 50 female, is used here. 800 images (about 8 samples per subject) of non-occluded frontal views with various facial expressions were used for training, while the others with sunglasses and scarves (as shown in Fig. 5) were used for testing. The images were resized to 83×60. To handle the occlusion, SRC uses l_1-norm to fit the coding error and the sparse coding model is: $(\hat{\boldsymbol{\alpha}}) = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_1 < \varepsilon$ [8]. Note that the use of l_1-norm on the coding error increases much the complexity of SRC.

The results are shown in Table 7. Although CRC_RLS is directly applied to the disguise face images, it gets the best result of FR with scarf disguise, outperforming SRC by a margin of 31%. For the case of FR with sunglasses, CRC_RLS is worse than SRC, but still better than SVM. We also partitioned the face image into 8 sub-regions for testing (the partition is the same as that in [8]). Then both the recognition rates of CRC_RLS and SRC are greater than 91%. These FR experiments with disguise again validate that CRC_RLS is very competitive.

Figure 5: The testing samples with sunglasses and scarves in the AR database.

Table 7: The results of face recognition with real disguise using the AR database.

	Sunglass	Scarf
SVM	66.5%	16.5%
SRC	**87.0%**	59.5%
SRC (partitioned)	**97.5%**	93.5%
CRC_RLS	68.5%	**90.5%**
CRC_RLS (partitioned)	91.5%	**95%**

In all the above FR experiments, both CRC_RLS and SRC are better than NN and LRC because of the benefit brought by CR. On the other hand, the result of CRC_RLS is comparable to SRC, showing that the l_1-norm regularization does not bring more benefit than the simple l_2-norm regularization in FR.

5.4. Running time

At last, let's compare the running time of CRC_RLS and SRC with various fast l_1-minimization methods, including l_1_*ls* [22], ALM [20], FISTA [19] and Homotopy[21]. We fix the dimensionality of Eigenface as 300. The recognition rates and speed of SRC and CRC_RLS are listed in Table 8 (Extended Yale B), Table 9 (AR) and Table 10 (Multi-PIE), respectively. Note that the results in Table 10 are the averaged values of Sessions 2, 3 and 4.

Table 8: Recognition rate and speed on the Extended Yale B database.

	Recognition rate	Time
SRC(l_1_ls)	**0.979**	5.3988 s
SRC(ALM)	**0.979**	0.128 s
SRC(FISTA)	0.914	0.1567 s
SRC(Homotopy)	0.945	0.0279 s
CRC_RLS	**0.979**	**0.0033 s**
Speed-up	**8.5 ~ 1636 times**	

Table 9: Recognition rate and speed on the AR database.

	Recognition rate	Time
SRC(l_1_ls)	0.933	1.7878 s
SRC(ALM)	0.933	0.0578 s
SRC(FISTA)	0.6824	0.0457 s
SRC(Homotopy)	0.8212	0.0305 s
CRC_RLS	**0.937**	**0.0024 s**
Speed-up	**12.6 ~ 744.9 times**	

Table 10: Recognition rate and speed on the MPIE database.

	Recognition rate	Time
SRC(l_1_ls)	**0.926**	21.2897 s
SRC(ALM)	0.9195	1.76 s
SRC(FISTA)	0.7955	1.636 s
SRC(Homotopy)	0.9017	0.5277 s
CRC_RLS	**0.922**	**0.0133 s**
Speed-up	**39.7 ~ 1600.7 times**	

On Yale B, CRC_RLS, SRC(l_1_*ls*) and SRC(ALM) achieve the best recognition rate (97.9%), but the speed of CRC_RLS is 1636 and 38.8 times faster than SRC(l_1_*ls*) and SRC(ALM). For the experiments on AR, CRC_RLS has the best recognition rate and speed. SRC(l_1_*ls*) is the second best but with the slowest speed. SRC(FISTA) and SRC(Homotopy) are much faster than SRC(l_1_*ls*) but they have lower recognition rates. On Multi-PIE, CRC_RLS achieves the second highest recognition rate (only 0.4% lower than SRC(l_1_*ls*)) but it is significantly (more than 1600 times) faster than SRC(l_1_*ls*). In this large-scale database, CRC_RLS is about 40 times faster than SRC with the fastest implementation (i.e., Homotopy) with more than 2% improvement in recognition rate.

From the results in the above three tests, we can see that the speed-up of CRC_RLS is more obvious as the scale (i.e., the number of classes or training samples) of face database increases. This implies that CRC_RLS is more advantageous in practical large-scale FR applications.

6. Conclusion and discussions

This paper revealed that it is the collaborative representation (CR) mechanism, but not the l_1-norm

sparsity constraint, that truly improves the face recognition (FR) accuracy. We then presented a very simple yet very effective FR scheme, namely CR based classification with regularized least square (CRC_RLS). Compared with the l_1-regularized sparse representation based classification (SRC), the l_2-regularized CRC_RLS has very competitive FR accuracy but with significantly lower complexity. The extensive experimental results clearly demonstrated that CRC_RLS is up to 1600 times faster than SRC without sacrificing recognition rate.

Apart from FR, our experiments on other types of signals (e.g., the human mouth odor signal classification for medical diagnosis) also showed that CRC or SRC works well. Statistically speaking, the norm (e.g., l_1 or l_2) imposed on the coding coefficient and coding error depends on the distributions of them (e.g., Laplacian or Gaussian). Nonetheless, more investigations are to be made to further study the CRC scheme for various pattern classification problems, and this is one of our main objectives in the future work.

References

[1] B. Olshausen and D. Field. Sparse coding with an overcomplete basis set: a strategy employed by V1? Vision Research, 37(23):3311–3325, 1997.

[2] M. Aharon, M. Elad, and A. M. Bruckstein. The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation. IEEE SP, 54(11):4311-4322, 2006.

[3] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, Non-local sparse models for image restoration. In ICCV 2009.

[4] R. Tibshirani. Regression shrinkage and selection via the LASSO. Journal of the Royal Statistical Society B, 58(1):267–288, 1996.

[5] D. Donoho. For Most Large Underdetermined Systems of Linear Equations the Minimal l_1-Norm Solution is also the Sparsest Solution. Comm. Pure and Applied Math., 59(6):797–829, 2006.

[6] J. A. Tropp and S. J. Wright. Computational methods for sparse solution of linear inverse problems. Proceedings of IEEE, Special Issue on Applications of Compressive Sensing & Sparse Representation, 98(6):948-958, 2010.

[7] K. Huang and S. Aviyente. Sparse representation for signal classification. In NIPS, 2006.

[8] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. IEEE PAMI, 31(2):210–227, 2009.

[9] S. H. Gao, I. W-H. Tsang, and L-T. Chia. Kernel Sparse Representation for Image Classification and Face Recognition. In ECCV, 2010.

[10] M. Yang and L. Zhang. Gabor Feature based Sparse Representation for Face Recognition with Gabor Occlusion Dictionary. In ECCV, 2010.

[11] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. Huang. Learning with l_1-graph for image analysis. IEEE IP, 19(4):858-866, 2010.

[12] J. Yang, K. Yu, Y. Gong and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In CVPR 2009.

[13] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang, and S. Yan. Sparse representation for computer vision and pattern recognition. Proceedings of IEEE, Special Issue on Applications of Compressive Sensing & Sparse Representation, 98(6):1031-1044, 2010.

[14] J. Z. Huang, X. L. Huang, and D. Metaxas. Simultaneous image transformation and sparse representation recovery. In CVPR 2008.

[15] A. Wagner, J. Wright, A. Ganesh, Z.H. Zhou, and Y. Ma, Towards a practical face recognition system: robust registration and illumination by sparse representation. In CVPR 2009.

[16] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. RASL: robust alignment by sparse and low-rank decomposition for linearly correlated images. Submitted to IEEE PAMI, 2010.

[17] A. Y. Yang, A. Ganesh, Z. H. Zhou, S. S. Sastry, and Y. Ma. Fast l_1-minimization algorithms and application in robust face recognition, UC Berkeley, Tech. Rep.

[18] S. J. Wright, R. D. Nowak, M. A. T. Figueiredo. Sparse reconstruction by separable approximation. In ICASSP, 2008.

[19] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM. J. Imaging Science, 2(1):183-202, 2009.

[20] J. Yang and Y. Zhang. Alternating direction algorithms for l_1-problems in compressive sensing. (*preprint*) *arXic:0912.1185*, 2009.

[21] D. Malioutove, M. Cetin, and A. Willsky. Homotopy continuation for sparse signal representation. In ICASSP, 2005.

[22] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. A interior-point method for large-scale l_1-regularized least squares. IEEE Journal on Selected Topics in Signal Processing, 1(4):606–617, 2007.

[23] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE PAMI, 23(6):643–660, 2001.

[24] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. IEEE PAMI, 27(5):684–698, 2005.

[25] A. Martinez, and R. benavente. The AR face database. CVC Tech. Report No. 24, 1998.

[26] R. Gross, I. Matthews. J. Cohn, T. Kanade, and S. Baker. Multi-PIE. Image and Vision Computing, 28:807–813, 2010.

[27] I. Naseem, R. Togneri, and M. Bennamoun. Linear regression for face recognition. IEEE PAMI, 32(11):2106-2112, 2010.

[28] G. H. Golub, and C. F. Van Loan, Matrix Computation, Johns Hopkins University Press, 1996.

[29] R. Rigamonti, M. Brown and V. Lepetit. Are Sparse Representations Really Relevant for Image Classification? In CVPR 2011.

[30] Q. Shi, A. Eriksson, A. Hengel, C. Shen. Is face recognition really a compressive sensing problem? In CVPR 2011.