

Integrating Simplified Inverse Representation and CRC for face recognition

Zhao Yingnan^{a,b,c}, Xiangjian He^c, Chen Beijing^{a,b}, Zhao Xiaoping^{a,b}

^aNanjing University of Information Science & Technology, Jiangsu Engineering Center of Network Monitoring, Nanjing, China, 210044

^bNanjing University of Information Science & Technology, School of Computer & Software, Nanjing, China, 210044

^cUniversity of Technology, Sydney, School of Computing and Communications, Sydney, Australia

Abstract Representation based classification method (RBCM) has attracted much attention in the last decade. RBCM exploits the linear combination of training samples to represent the test sample, which then be classified according to the minimum reconstruction residual. Recently, an interesting concept, Inverse Representation (IR), is proposed. It is the inverse process of the conventional RBCM, which applies test samples' information to represent each training sample, and then classify the training sample as a useful supplementary for the final classification. The relative algorithm called CIRLRC, integrating IR and linear regression classification (LRC) by score fusing, does show the superior classification performance. However, there are two main drawbacks in CIRLRC. One it is not a pure IR, for the test vector contains some training sample information. The other is the computation inefficiency that CIRLRC should solve C linear equations for classifying the test sample respectively, where C is the number of the classes. Therefore, we present a novel method integrating simplified IR(SIR) and collaborative representation classification (CRC) for face recognition (SIRCRC). In SIRCRC, only test sample information is fully used in SIR, and CRC is more efficient than LRC in terms of speed, that is, 1 linear equation system is needed. Extensive experimental results on face databases show that it is very competitive with both CIRLRC and the state-of-the-art RBCM.

Keywords: face recognition, inverse representation, collaborate recognition classification

1 Introduction

Representation based classification method (RBCM) has emerged as a powerful tool in a wide range of application fields, especially in signal processing[1,2], image processing[3-5] and visual tracking[6, 7]. Also, it plays an important role in biometrics recognition, such as face[8-12], palmprint[13], ear[14], fingerprint[15] and iris[16]. RBCM requires a test sample to be sparsely represented by a weighted sum of all the training samples. According to the type of norm regularization, RBCM can be classified into three categories, that is, l_1 norm, l_2 norm and l_p norm($0 < p < 1$).

It is reported that l_0 norm can measure the sparsity, while it is not tractable due to NP-hard property. As the closest convex function to l_0 norm, l_1 minimization is widely used in RBCM. Sparse Representation based Classification (SRC) [8] can be referred as the milestone of RBCM in face recognition. In SRC, the query face image is coded as a sparse linear combination of all the training samples via l_1 norm minimization. Not only it yields high recognition accuracy in face databases, but also it is robust to face occlusion and corruption. The success of SRC boots the research of sparsity based pattern classification, and the extensive RBCM with l_1 norm minimization have been proposed, such as sparse coding for face recognition[17] and sparse image classification [18]. However, some researchers doubt about the role of sparsity in face recognition[9]. Lei Zhang et. al. pointed out that it is collaborate representation (CR, i.e. representing the query image collaboratively by samples from all the classes) not l_1 norm sparsity to contribute the final classification accuracy. The non-sparse l_2 norm to calculate the representation coefficients could lead to similar recognition but significantly high computational speed. Based on l_2 norm CR, they proposed a simple but more efficient collaborate representation classification (CRC). But it is noted that all the feature elements both in SRC and CRC share the same coding vector over their associated sub-dictionaries. This requirement ignores the fact that the feature elements in a pattern not only share similarities but also have differences. Therefore, Meng Y. et. al. present relaxed collaborative representation (RCR)[11] model to effectively exploit the similarity and distinctiveness of features. Also, Linear regression classification (LRC)[12] can be referred as a l_2 norm based on the linear regression model. In addition to the l_1 norm and l_2 norm minimization, some researchers are trying to solve the sparse representation problem with the l_p norm ($0 < p < 1$), especially $p = 0.1, \frac{1}{2}, \frac{1}{3}$ or 0.9 [19-21]. More information about RBCM can be found in the review of Ref.[22].

More Recently, an interesting concept, Inverse Representation (IR)[23], is proposed. In essence, IR is the inverse process of RBCM, i.e. it classifies the training sample using the test sample's information. The relative algorithm CIRLRC is integrating IR and linear regression classification (LRC) by score fusing. It does show the superior classification performance. However, there are two main drawbacks in CIRLRC. One it is not a pure IR, for the test vector contains some training sample information. The other is the computation inefficiency that CIRLRC should solve C linear equations for classifying the test sample respectively, where C is the number of the classes. Therefore, we present a novel method integrating simplified IR(SIR) and collaborative representation classification (CRC) for face recognition (SIRCRC). In SIRCRC, only test sample information and its mirror images are fully used in SIR, and CRC is more efficient than LRC in terms of speed, that is, 1 linear equation system is needed.

The rest of this paper is organized as follows. Section 2 gives a brief review of some related works. Section 3 presents the proposed SIRCRC method and some analysis. Section 4 performs experiments and Section 5 concludes the paper.

2 Review of related works

2.1 Presentation of CIRLRC[23]

The proposed CIRLRC in Ref.[23] exploits the conventional LRC and IR to generate two kinds of scores, and then combine them by score fusing to recognize the face. The comparison of LRC and CRC presents in Section 2.2. In CIRLRC, the IR is the inverse process of LRC and mirror samples are used to form the visual samples. Assume that there are C classes and each class has n training samples. We denote $X = [x_1, \dots, x_N]$ as all the N training samples, and $y \in Y$ as the test sample. The main steps in CIRLRC is as follows:

- Step 1 Mirror samples creation. Calculate the mirror samples[24] of training samples x and test sample y , respectively, denoted as x^v and y^v .
- Step 2 LRC process. Let the training sample vector contains original samples and its mirror images, denoted as $X' = [x_1 \dots x_n x_1^v \dots x_n^v \dots x_{(C-1)n+1} \dots x_N x_{(C-1)n+1}^v \dots x_N^v]$ ($N = Cn$). The linear system is $y = X_i \alpha_i$, $i = 1, \dots, C$. Using LRC method to classify each test sample and yield one score named S_{LRC} .
- Step 3 IR process.
- Test vector definition. For the j -th class, combine all the naive training samples from the other classes, the naive test sample and virtual test sample to form the test vector Z , where $Z = [X'_1 \dots X'_{j-1} \quad X'_{j+1} \dots X'_C \quad y \quad y^v]$.
 - IR classification. Z acts as the training vector and each naive training sample from the j -th class acts as the test sample. Then apply inverse LRC to classify each naive training sample from the j -th class. Here, the residual is calculated using $d_k^0 = \|X - \hat{\beta}_k^{(C-1)n+1} y - \hat{\beta}_k^{(C-1)n+2} y^v\|$. For the j -th class, the mean of d_k^0 is used as the distance between the test sample and the j -th class and is denoted by d_j . It yields the other score named S_{IRLRC} .
- Step 4 Score fusing. The final score is calculated by $S = \omega_1 S_{LRC} + \omega_2 S_{IRLRC}$, where ω_1 and ω_2 are fusing coefficients, and $\omega_1 + \omega_2 = 1$. According to the final score, classify the test sample finally.

We can observe that the test vector contains some training sample information in Step 3. Therefore it is not a pure IR. In the proposed SIRCRC, we simplify the IR definition, which contains only test sample information, which have a better classification accuracy.

2.2 Comparison of LRC and CRC

In this subsection, we present the comparison of LRC and CRC. Table 1 and Table 2 shows the LRC and CRC algorithms respectively.

Table 1: The LRC Algorithm

1. $\hat{\beta}_i \in R^{p_i \times 1}$ is evaluated against each class model,

$$\hat{\beta}_i = (X_i^T X_i)^{-1} X_i^T y, i = 1, 2, \dots, C \quad (1)$$

2. Calculate the distance between original and predicted response variable

$$d_i(y) = \|y - X_i \hat{\beta}_i\|_2, i = 1, 2, \dots, C \quad (2)$$

3. If $k = \arg \min_i d_i(y)$, then the test sample is assigned to the k -th class.
-

Table 2: The CRC Algorithm

1. Normalize the columns of X to have unit l_2 -norm.

2. Code y over by

$$\hat{\rho} = (X^T X + \lambda I)^{-1} X^T y \quad (3)$$

3. Compute the regularized residuals

$$r_i = \|y - X_i \hat{\rho}\|_2, i = 1, 2, \dots, C \quad (4)$$

4. Output the identity of y as

$$\text{Identity}(y) = \arg \min_i \{r_i\}.$$

During the residual calculation process, both LRC and CRC fall into the l_2 minimization category according to Eq. (2) and Eq. (4). We can observe from the tables, that the main difference lies in Eq. (1) and Eq. (3). Indeed, LRC and CRC are based on the different ideas. The former is based on the linear regression model, and the latter is the collaborate representation. But, they can all be attributed to the minimum squared error (MSE) problem. As for Eq. (1), in general, we use the following to make the least square solution stable and to impose a weaker sparsity constraint on the solution:

$$\hat{\beta}_i = (X_i^T X_i + \lambda I)^{-1} X_i^T y, i = 1, 2, \dots, N \quad (5)$$

where λ is a positive constant, which is the same in Eq. (3). However, in Eq. (1), LRC should solve C linear equations for classifying the test sample respectively, where C is the number of the classes. And Eq. (3) in CRC is more efficient than LRC in terms of speed, that is, one linear equation system is needed. Therefore, we prefer CRC in the proposed method.

3 Description of proposed method SIRCRC

3.1 SIRCRC Framework

In this subsection, we illustrate the proposed SIRCRC in detail. Comparing to the CIRCRC, we make two improvements. One is the simplification of IR definition, which is more reasonable

and leads to higher classification accuracy. The other lies in the integrating the presented SIR to CRC instead of LRC, which produces higher computation speed. It is well known that the number of training samples is always bigger than that of test samples in real applications. When we try to consider the test vector as the training samples according to the IR definition, we need to enlarge its number. In other words, we need to construct virtual test samples. Mirror image is a simple but effective method[24]. Let $X = [x_1, \dots, x_N]$ stand for the training sample vector, and $Y = [y_1, \dots, y_M]$ is the test sample vector respectively. Suppose that there are C classes and each class has n training samples and m testing samples. Obviously, $N = Cn$ and $M = Cm$. $x_{(i-1)n+k}$ stands for the k -th training sample of the i -th class. Similarly, $y_{(j-1)m+p}$ stands for the p -th training sample of the j -th class.

The proposed method is described as follows. First, it forms the test vector by test sample and its mirror image. Note that all the test sample information is in the test vector, no training information at all, not like CIRLRC. We think it is more natural according to the IR essence. Further, for score fusing, more uncorrelated in each item, better recognition rate it yields. SIR definition is more independent than IR. Second, we obtain the optimal linear training samples from every class to represent the test sample, and calculates the score of each class. Here the classical CRC method is applied. Third, we conduct SIR on the base of CRC. Finally, it fuses the scores produced from the second and third steps for the ultimate classification. These steps are presented in detail below.

Step 1 Produce the test vector. It contains two parts, original test sample and mirror images. For original test sample y in the form of face image, the virtual test sample is defined as

$$y^v(t, s) = y(t, S - s + 1), \quad t = 1, \dots, T, \quad s = 1, \dots, S \quad (6)$$

where T is the rows and S is the columns of the face image matrix. The relationship between y and y^v is that column vector y^v is obtained by concatenating the rows of y in sequence.

Step 2 CRC procedure. We first establish the linear system

$$y = XA \quad (7)$$

In order to simultaneously minimize the norm of the solution vector, we give an objective function as $\min y = \|XA\| + \lambda\|A\|$. Hereafter $\|\cdot\|$ always denotes the l_2 norm.

Therefore, we solve A using

$$\hat{A} = (X^T X + \lambda I)^{-1} X^T Y \quad (8)$$

λ and I stand for a small positive constant and the identity matrix, respectively. We calculate the score between the test sample y and the i -th class using $s_i = \|y - X_i \hat{A}\|$.

Step 3 SIR procedure. Define the test vector Z first.

$$Z=[Y \quad Y^v] \quad (9)$$

Then we establish a linear system for each naive training sample as $X=Z\beta$. β is solved using

$$\hat{\beta}=(Z^T Z+\lambda I)Z^T X \quad (10)$$

λ and I still stand for a small positive constant and the identity matrix, respectively.

We use the following

$$d_i=\|X_i-\hat{\beta}Z\| \quad (11)$$

as the distance of between the test sample X and the training sample Z .

Step 4 Score fusing. For test sample y , we first normalize its scores and "distance" with respect to all the classes using

$$s'_j=(s_j-s^{\min})/(s^{\max}-s^{\min}) \quad (12)$$

$$d'_j=(d_j-d^{\min})/(d^{\max}-d^{\min}) \quad (13)$$

where $s^{\min}=\min(s_1,\dots,s_C)$, $s^{\max}=\max(s_1,\dots,s_C)$, $d^{\min}=\min(d_1,\dots,d_C)$, $d^{\max}=\max(d_1,\dots,d_C)$. It uses $t_j=\omega_1 s'_j+\omega_2 d'_j$ to calculate the ultimate score with respect to the j -th class. ω_1 and ω_2 are the weights and $\omega_1+\omega_2=1$. Because conventional representation seems to be more reliable in evaluating the dissimilarity than the SIR, we often assigns a larger value to ω_1 in comparison to ω_2 . If $k=\arg\min_j t_j$, then test sample y is assigned to the k -th class.

3.2 Analysis of SIRCRC

The proposed SIRCRC method has two main contributions. One is that the simplification of IR. The other is that we apply CRC instead of LRC.

We can observe from Step 3 in the original CIRLRC that the test vector contains three parts: original training sample, the test sample and its mirror images. In the proposed SIRCRC, we simplify it by getting rid of the naive training sample. The essence of IR is the inverse procedure of conventional RBCM. Hereafter, in SIRCRC we make the best of the test sample information. Another drawback of IR definition in CIRLRC lies in the distance calculation phase. Owing to the training sample part in the test vector of IR, we couldn't calculate them to the naive training sample. Therefore, CIRLRC calculate the other two parts respectively and then average them. As for SIR in SIRCRC, all the information in the test vector is about test sample, we can compute the residual difference directly. It is natural, simple and efficient.

As it has analyzed in CIRLRC, in real-world applications, the error exists in both the test sample and the training one. The conventional RBCM is based on the least-squares algorithm, which takes only the error in the test sample into account. Actually, Eq. (7) can be rewritten as

$$XA=y=y_0+\Delta y \quad (14)$$

where y_0 and Δy stand for the true test sample and error, respectively. And it generates the

following objective function:

$$\{\hat{A}, \Delta y\} = \operatorname{argmin}(\|\Delta y\| + \lambda \|A\|) \quad s.t. XA = y_0 + \Delta y \quad (15)$$

While the IR takes only the error in the training sample into account. The equation $Z\beta = x$ can be rewritten as

$$Z\beta = x = x_0 + \Delta x \quad (16)$$

where x_0 and Δx stand for the true train sample and error, respectively. It is easy to know that the relative objective function is as follows:

$$\{\hat{\beta}, \Delta x\} = \operatorname{argmin}(\|\Delta x\| + \lambda \|\beta\|) \quad s.t. Z\beta = x_0 + \Delta x \quad (17)$$

We see from the above presentation that it allows the error both in the test sample and the training sample to be considered and processed simultaneously. This will be beneficial to achieve good face recognition performance. However, Eq. (16) takes account of error in the training sample on the right side, but the error of the training sample in the test vector is neglected. When we use the proposed SIR, this drawback will be tackled.

Both CIRLRC and the presented SIRCRC adopt the integrating technique and score fusing finally. In score level fusion, if correlation coefficient between the two kinds of scores to fuse is low, the fusion result is usually good. That is to say, a smaller correlation coefficient allows the fusion to better accuracy. Let x and y be two variables. Generally, we define the correlation coefficient between them as

$$\rho(x, y) = \frac{\operatorname{cov}(x, y)}{\sqrt{\operatorname{cov}(x, x)}\sqrt{\operatorname{cov}(y, y)}} \quad (18)$$

where $\operatorname{cov}(x, y) = E[(x - E(x))(y - E(y))]$. As for the two residuals s and d from Step 2 and Step 3 respectively, we calculate the correlation coefficient between them, that is

$$\rho(s, d) = \frac{\operatorname{cov}(s, d)}{\sqrt{\operatorname{cov}(s, s)}\sqrt{\operatorname{cov}(d, d)}} \quad (19)$$

where $\operatorname{cov}(s, d) = \frac{1}{C} \sum_{j=1}^C (s_j - \bar{s})(d_j - \bar{d})$, $\bar{s} = \frac{1}{C} \sum_{j=1}^C s_j$, $\bar{d} = \frac{1}{C} \sum_{j=1}^C d_j$.

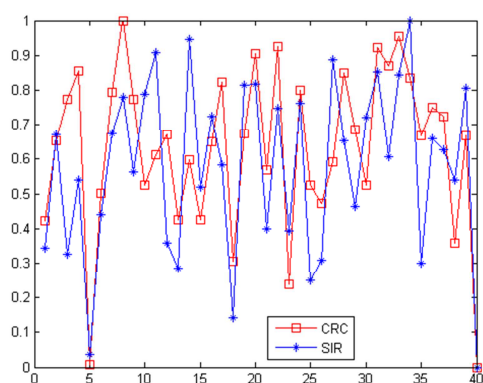
Table 1 shows the mean of all the correlation coefficients of the scores and distances of the test samples from the ORL database. The highest mean of correlation coefficients of CIRLRC is 0.5469, while that in SIRCRC is 0.4939. It implies that SIRCRC has lower difference than CIRLRC, and hence the better fusion performance.

Fig. 1 depicts the scores and distances of the last test sample, obtained using Steps 2 and 3 in the case where the first 5 face images of each subject in the ORL database are used as training samples and the others are taken as test samples. Fig.1(a) is SIRCRC and Fig.1(b) is CIRLRC. We can observe that in the former there are 3 points with similar data, while the latter has 9 such points. Hence, for SIRCRC, the difference of the correlation coefficient of scores in CRC and distance in SIR is greater than that in CIRLRC. Fig. 2 depicts the scores and distances, of the last

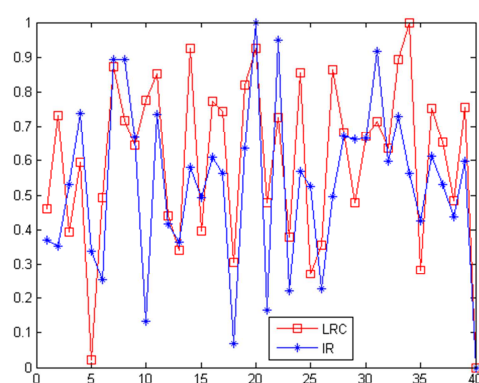
test sample from the subset of the FERET database, obtained using Steps 2 and 3 in the case where the first 4 images of each subject are used as training samples and the rest as the test samples. We can easily draw the similar conclusion to Fig. 1.

Table 1 Mean of all the correlation coefficients of the residuals from Step 2 and Step 3 of CIRLRC and the proposed SIRCRC in ORL face database.

Training sample per class	3	4	5
CIRLRC	0.5469	0.5266	0.5024
SIRCRC	0.4939	0.4863	0.4733

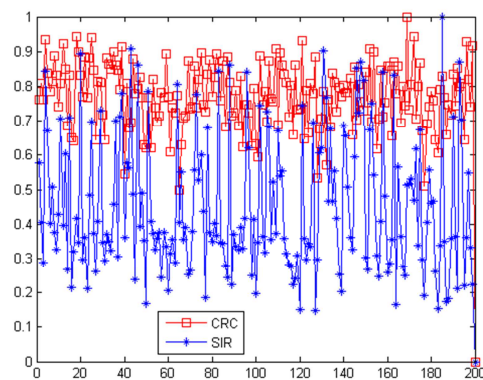


a. SIRCRC

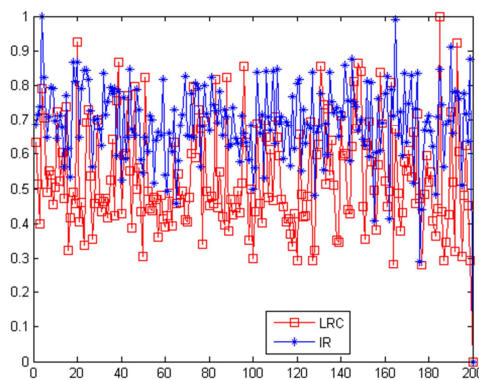


b. CIRLRC

Fig. 1 Residuals of the last test sample obtained using steps 2 and 3 of the methods. Fig. 1(a) is SIRCRC and Fig. 1(b) is CIRLRC. The first 5 images of each subject in the ORL database are used as training samples and the others are taken as test samples. The vertical axis shows the values of residuals and the horizontal axis shows the no. of the component of the normalized score vector and distance vector.



a. SIRCRC



b. CIRLRC

Fig. 2 Residuals of the last test sample obtained using steps 2 and 3 of the methods. Fig.2 (a) is SIRCRC and Fig. 2 (b) is CIRLRC. The first 4 images of each subject in the FERET database are used as training samples and the others are taken as test samples. The vertical axis shows the values of residuals and the horizontal axis shows the no. of the component of the normalized score vector and distance vector.

4 Experiments

We use the ORL, FERET and Georgia Tech(GT) databases to test the presented SIRCRC. Fig. 3-5 gives some samples of these face databases. We compare it with CIRLRC[23], and some state-of-art face recognition methods as well, such as SRC[8], LRC[12], CRC[9]and RCR[11]. In SIRCRC, we guarantee the two weights $\omega_1 + \omega_2 = 1$. Therefore, we just show the value of ω_1 . While in CIRLRC, we only adopt the optimal weights. And the parameter λ in Eq.(8) and Eq.(10) is set to 0.001.



Fig. 3. Some samples from ORL face database.



Fig. 4. Some samples from FERET.

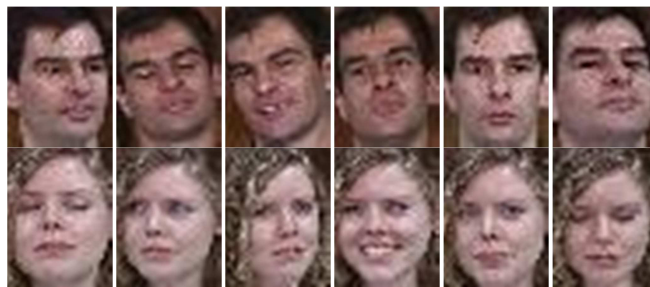


Fig. 5. Some samples from GT face database.

4.1 Experiments on the ORL face database

We use the ORL face database [24] to evaluate our method. There are 400 gray images from 40 subjects. Every subject provides 10 images. For some subjects, the images were taken at different times, with varying lighting, facial expressions and facial details. Each image was also resized to an image with one half of the original size by using the down-sampling algorithm. We respectively take the first 2, 3 and 4 face images of each subject as original training samples and

treated the remaining face images as test samples. The experimental results are shown in Table 2. It shows that both CIRLRC and SIRCRC are able to perform better than the classical RBCM, such as SRC, RCR, and LRC and CRC as well. In the row of CIRLRC, we also give the recognition error rates of LRC and CIR, that is the two parts of score fusing. In the same way, we list them, CRC and SIR, in the SIRCRC. Compare CRC and LRC, we can observe that CRC performs well on the ORL face database. And SIR is not as well as CIR in terms of recognition performance. While as we have analyzed in Sec. 3.2, the two parts in SIRCRC has greater irrelevance than in CIRLRC. Hereafter, we get a lower final error rates than CIRLRC on the ORL face database.

Table 2 Recognition error rates (%) of different methods on the ORL face database

Number of the original training samples per class	2	3	4
SRC	22.25	11.93	09.92
RCR	21.77	18.86	17.52
LRC	21.56	18.93	15.42
CIR	20.26	18.57	17.92
CIRLRC($\omega_1=0.6$)	09.64	08.33	11.00
CRC	16.56	15.00	11.25
SIR	21.38	20.43	18.33
SIRCRC($\omega_1=0.8$)	09.93	08.18	07.33
SIRCRC($\omega_1=0.7$)	09.56	07.94	08.15
SIRCRC($\omega_1=0.6$)	08.60	07.49	09.50

4.2 Experiments on the FERET face database

The FERET database is one of the standard facial image database specially used for the face recognition algorithms[25]. We use a subset of it. It is composed of 1400 images from 200 individuals with each subject providing 7 images. This subset includes the face images whose names contain two-character string: “ba”, “bj”, “bk”, “be”, “bf”, “bd” and “bg”. The images in this subset have pose variations of $\pm 15^\circ$, $\pm 25^\circ$, and also the variations of the illumination and expression. Before experiment, we use the down-sampling algorithm to resize each image into a 40×40 pixel. Table 3 shows that our proposed method usually classifies more accurately than CIRCLC and the classical RBCM.

Table 3 Recognition error rates (%) of different methods on the FERET face database

Number of the original training samples per class	1	2	3
SRC	64.90	52.77	56.00
RCR	79.99	80.18	88.52
LRC	80.18	59.97	60.87
CIR	81.47	61.23	63.83
CIRLRC($\omega_1=0.8$)	64.39	53.97	54.77
CRC	55.67	41.60	55.63
SIR	57.75	49.60	53.25
SIRCRC($\omega_1=0.8$)	52.83	37.20	43.13
SIRCRC($\omega_1=0.7$)	52.58	38.30	45.62
SIRCRC($\omega_1=0.6$)	50.83	36.50	43.13

4.3 Experiments on the GT face database

In this subsection, we use the Georgia Tech (GT) face database [33] to test our method. It was built at Georgia Institute of Technology, which contains images of 50 people taken in two or three sessions. All people in the database were represented by 15 color JPEG images with cluttered background taken at the resolution of 640×480 pixels. The pictures show frontal or tilted faces with different facial expressions, lighting conditions and scale. Each image was manually labeled to determine the position of the face in the image. We use the face images with the background removed and each of these face images has the resolution of 40×30 pixels. They are all converted into gray images in advance. The first 2, 3 and 4 face images of each subject are used as training samples and the remaining images are taken as test samples. Table 4 shows again that CIRLRC and SIRCRC are better than SRC, RCR, LRC and CRC. Meanwhile, CIRLRC and SIRCRC have the similar recognition error rates.

Table 4 Recognition error rates (%) of different methods on the GT face database

Number of the original training samples per class	2	3	4
SRC	45.26	42.17	40.24
RCR	63.71	61.57	58.21
LRC	54.15	49.17	44.36
CIR	64.15	64.33	62.55
CIRLRC($\omega_1=0.9$)	46.92	42.83	39.45
CRC	57.54	54.50	52.55
SIR	58.31	57.50	55.09
SIRCRC($\omega_1=0.8$)	45.69	43.50	40.00
SIRCRC($\omega_1=0.7$)	45.23	42.33	40.18
SIRCRC($\omega_1=0.6$)	45.54	41.50	48.91

4.4 Running time

In this subsection, we compare the computation efficiency between CIRLRC and SIRCRC on the ORL, FERET and GT face databases. The first 5, 3 and 4 face images of each subject are used as training samples and the remaining images are taken as test samples in the ORL, FERET and GT respectively. Table 5 describes the speed of CIRLRC and SIRCRC on face databases. It is obvious that the presented SIRCRC is more computational efficient than CIRLRC by the 11.81 averaging speed-up times.

Table 5 Speed on the face databases (Time s)

	Number of training samples per class	CIRLRC	SIRCRC	Speed-up(times)
ORL	5	1452.51	139.93	10.38
FERET	3	16495.38	1138.77	14.49
GT	4	602.26	57.11	10.55

5 Conclusions

Representation based classification method (RBCM) has attracted much attention in the last decade. It exploits the linear combination of training samples to represent the test sample and

then classify the test sample according to the minimum reconstruction residual. Among the RBCM, a novel concept, Inverse Representation (IR), is proposed recently. It makes the most of test samples' information to represent each training sample. The relative CIRLRC algorithm integrates IR and LRC by score fusing and shows the superior classification performance. However, it suffers from two aspects. One it is that the test vector contains some training sample information. The other is the computation inefficiency that CIRLRC should solve C linear equations for classifying the test sample respectively, where C is the number of the classes. Therefore, we present a novel method integrating simplified IR(MIR) and collaborative representation classification (CRC) for face recognition (SIRCRC). In SIRCRC, only test sample information is fully used in SIR, and CRC is more efficient than LRC in terms of speed, that is, 1 linear equation system is needed. Extensive experimental results on ORL, FERET and GT databases show that it is very competitive with both CIRLRC and the state-of-the-art RBCM.

Acknowledgments

This work is supported in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions, Natural Science Foundation of China (No. 61103141) and Student Innovation Training Program of NUIST(No. 201410300190 and No. 201410300178)

References

1. Yuangqing Li, Liangyu Zhu, Ning Bi, and Yong Xu, Sparse representation for brain signal processing: A tutorial on methods and applications, *Signal Processing*, vol. 31, no. 3, pp. 96-106, 2014.
2. Boufounos P., Kutyniok G., and Rauhut H., Sparse recovery from combined fusion frame measurements, *IEEE Trans. on Information Theory*, vol. 57, no. 6, pp. 3864-3876, 2011.
3. Arbelaez, Pablo, Michael Maire, Charless Fowlkes, and Jitendra Malik. "Contour detection and hierarchical image segmentation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33, no. 5 , pp. 898-916, 2011.
4. Yang, Jianchao, John Wright, Thomas S. Huang, and Yi Ma. "Image super-resolution via sparse representation." *Image Processing, IEEE Transactions on* 19, no. 11: 2861-2873, 2010.
5. Mairal, Julien, Michael Elad, and Guillermo Sapiro. "Sparse representation for color image restoration." *Image Processing, IEEE Transactions on* 17, no. 1: 53-69, 2008.
6. Mei, Xue, and Haibin Ling. "Robust visual tracking and vehicle classification via sparse representation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33, no. 11 (2011): 2259-2272.

7. Jia, Xu, Huchuan Lu, and Ming-Hsuan Yang. "Visual tracking via adaptive structural local sparse appearance model." In *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*, pp. 1822-1829. IEEE, 2012.
8. Wright, John, Allen Y. Yang, Arvind Ganesh, Shankar S. Sastry, and Yi Ma. "Robust face recognition via sparse representation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31, no. 2 (2009): 210-227.
9. Zhang, D., Meng Yang, and Xiangchu Feng. "Sparse representation or collaborative representation: Which helps face recognition?." In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 471-478. IEEE, 2011.
10. Xu, Yong, David Zhang, Jian Yang, and Jing-Yu Yang. "A two-phase test sample sparse representation method for use with face recognition." *Circuits and Systems for Video Technology, IEEE Transactions on* 21, no. 9 (2011): 1255-1262
11. Yang, Meng, D. Zhang, and Shenlong Wang. "Relaxed collaborative representation for pattern classification." In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2224-2231. IEEE, 2012.
12. .Naseem, Imran, Roberto Togneri, and Mohammed Bennamoun. "Linear regression for face recognition." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32, no. 11 (2010): 2106-2112.
13. Xu, Yong, Zizhu Fan, Minna Qiu, David Zhang, and Jing-Yu Yang. "A sparse representation method of bimodal biometrics and palmprint recognition experiments." *Neurocomputing* 103 (2013): 164-171.
14. Naseem, Imran, Roberto Togneri, and Mohammed Bennamoun. "Sparse representation for ear biometrics." In *Advances in visual computing*, pp. 336-345. Springer Berlin Heidelberg, 2008.
15. Shekhar, Sumit, Vishal M. Patel, Nasser M. Nasrabadi, and Rama Chellappa. "Joint sparse representation for robust multimodal biometrics recognition." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 36, no. 1 (2014): 113-126.
16. Pillai, Jaishanker K., Vishal M. Patel, Rama Chellappa, and Nalini K. Ratha. "Secure and robust iris recognition using random projections and sparse representations." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33, no. 9 (2011): 1877-1893.
17. Yang, Jianchao, Kai Yu, Yihong Gong, and Thomas Huang. "Linear spatial pyramid matching using sparse coding for image classification." In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1794-1801. IEEE, 2009.
18. Yang, Meng, D. Zhang, and Jian Yang. "Robust sparse coding for face recognition." In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 625-632. IEEE, 2011.
19. Lyu, Qin, Zhouchen Lin, Yiyuan She, and Chao Zhang. "A comparison of typical l_p minimization algorithms." *Neurocomputing* 119 (2013): 413-424.

20. Xu, Zongben, Xiangyu Chang, Fengmin Xu, and Hai Zhang. " $l_{1/2}$ regularization: A thresholding representation theory and a fast solver." *Neural Networks and Learning Systems, IEEE Transactions on* 23, no. 7 (2012): 1013-1027.
21. Guo, Song, Zhan Wang, and Qiuqi Ruan. "Enhancing sparsity via l_p minimization for robust face recognition." *Neurocomputing* 99 (2013): 592-602.
22. Zhang, Zheng, Yong Xu, Jian Yang, Xuelong Li, and David Zhang. "A survey of sparse representation: algorithms and applications.", DOI: 10.1109/ACCESS.2015.2430359
23. Xu, Yong, Xuelong Li, Jian Yang, Zhihui Lai, and David Zhang. "Integrating conventional and inverse representation for face recognition." *Cybernetics, IEEE Transactions on* 44, no. 10 (2014): 1738-1746.
24. Available: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>.
25. Available: <http://www.itl.nist.gov/iad/humanid/feret>.
26. Available: http://www.anefian.com/research/face_reco.html.