The Hong Kong Polytechnic University

Department of Computing

Regularized Robust Coding and Dictionary Learning for Face Recognition

by

YANG Meng

A thesis submitted in partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

April 2012

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree of diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

YANG Meng (Name of Student)

Abstract

How to represent the object and how the object representation should be learnt are very fundamental problems in pattern classification tasks, for example, face recognition (FR). As one of the most visible research topics in computer vision, machine learning and biometrics, robust FR to occlusions, misalignment and various variations (e.g., pose, expression and illumination) is still a very challenging problem after many years' investigation. Recently, the sparse representation theory has been rapidly developed and successfully used in solving various inverse problems such as image reconstruction. Efforts have also been made in using sparse representation for signal classification. In particular, by coding a testing face sample as a sparse linear combination of the training samples and classifying it by evaluating which class leads to the minimum coding residual, sparse representation based classification (SRC) leads to very interesting results for FR. The success of SRC greatly boosts the research of sparsity based classification and the associated dictionary learning techniques.

Though SRC has shown promising performance in robust FR, there are still many problems to be further addressed. What is the working mechanism of SRC? What is the role of l_0 or l_1 norm sparsity in it? How to extract effective features to improve the accuracy and speed of SRC? How to design a robust representation fidelity term to handle various outliers? How to train a dictionary to improve classification? In this thesis, we aim to answer these questions with tools from statistical learning, convex optimization, and pattern classification.

It is widely believed that the l_1-norm sparsity constraint on the coding coefficients plays a key role in the success of SRC. In this thesis, however, it is shown that the collaborative representation mechanism (i.e., using all training samples to collaboratively represent the testing sample) is much more crucial than the l_1-norm sparsity of coding coefficients to the success of face classification. A new framework, namely collaborative

representation based classification (CRC), is then established and discussed conceptually and experimentally. CRC has various instantiations by applying different norms to the coding residual and coding coefficient, while SRC is a special case of it. It is further shown that l_2-regularizatoin of coding coefficients in CRC could achieve similar performance to or better performance than l_1-regularization and have higher computational efficiency.

We then discuss the use of local features to improve the performance and speed of SRC. We present a Gabor feature based robust representation and classification (GRRC) scheme with Gabor occlusion dictionary (GOD) learning. It is shown that the use of Gabor feature and GOD not only improves the FR accuracy but also reduces significantly the computational cost in handling face occlusion. This part of work also indicates that the appropriate representation model (e.g., the regularization and dictionary) has a close relationship to the feature of the involved signals, which should be considered in designing effective representation models.

The third major contribution of this thesis is the development of regularized robust coding (RRC) for FR. In RRC, a robust representation fidelity term is proposed to handle various outliers in face images. RRC is a *maximum a posterior* solution by assuming that the coding residual and the coding coefficient are respectively independent and identically distributed. An iteratively reweighted regularized robust coding algorithm is developed to solve the RRC model efficiently. Extensive experiments on representative face databases demonstrate that the RRC is much more effective and efficient than state-of-the-art sparse representation based methods in dealing with face occlusion, corruption, lighting and expression changes, etc.

Finally, we discuss the problem of dictionary learning (DL) for sparse representation based pattern classification, and propose a novel Fisher discrimination dictionary learning (FDDL) scheme. Based on the Fisher discrimination criterion, a structured dictionary, whose dictionary atoms have correspondence to the class labels, is learnt so that the reconstruction residual after sparse coding can be used for pattern classification. Meanwhile, the Fisher discrimination criterion is imposed on the coding coefficients so

that they have small within-class scatter but big between-class scatter. A new classification scheme associated with the proposed FDDL method is then presented by using both the discriminative information in the reconstruction residual and sparse coding coefficient. The proposed FDDL is extensively evaluated on benchmark image databases in comparison with existing sparsity and DL based classification methods.

Publications

The following papers, published, in press or submitted, are the partial outputs of my PhD studies in PolyU.

Conference papers:

1. **Meng Yang**, Lei Zhang and David Zhang, "Misalignment Robust Representation for Face Recognition," *Proceedings of the 12th European Conference on Computer Vision (ECCV)*, 2012.
2. **Meng Yang**, Lei Zhang, David Zhang, and Shenlong Wang, "Relaxed Robust Representation for Pattern Classification," *Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
3. **Meng Yang**, Lei Zhang, Xiangchu Feng and David Zhang, "Fisher Discrimination Dictionary Learning for Sparse Representation," *Proceedings of the 13th International Conference on Computer Vision (ICCV)*, 2011.
4. Lei Zhang, **Meng Yang**, and Xiangchu Feng, "Sparse representation or collaborative representation which helps face recognition?" *Proceedings of the 13th International Conference on Computer Vision (ICCV)*, 2011.
5. **Meng Yang**, Lei Zhang, Jian Yang and David Zhang, "Robust Sparse Coding for Face Recognition," *Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.
6. **Meng Yang** and Lei Zhang, "Gabor Feature based Sparse Representation for Face recognition with Gabor Occlusion Dictionary," *Proceedings of the 11th European Conference on Computer Vision (ECCV)*, 2010.
7. **Meng Yang**, Lei Zhang, Lin Zhang, and David Zhang, "Monogenic binary pattern (MBP): a novel feature extraction and representation model for face recognition," *Proceedings of the 20th International Conference on Pattern Recognition* (ICPR),

2010.

8. **Meng Yang**, Lei Zhang, Jian Yang and David Zhang, "Metaface Learning for Sparse Representation based face recognition," *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP)*, 2010.
9. Lei Zhang, **Meng Yang**, Zhizhao Feng, and David Zhang, "On the Dimensionality Reduction for Sparse Representation based Face Recognition," *Proceedings of the 20th International Conference on Pattern Recognition (ICPR)*, 2010.

Journal papers:

1. **Meng Yang**, Lei Zhang, Simon C. K. Shiu, and David Zhang, "Gabor Feature based Robust Representation and Classification for Face Recognition with Gabor Occlusion Dictionary," accepted by *Pattern Recognition*.
2. **Meng Yang**, Lei Zhang, Jian Yang, and David Zhang, "Regularized Robust Coding for Face Recognition," *IEEE Transactions on Image Processing* (under major revision).
3. **Meng Yang**, Lei Zhang, Simon C. K. Shiu, and David Zhang, "Monogenic Binary Coding: An Efficient Local Feature Extraction Approach to Face Recognition," accepted by *IEEE Transactions on Information Forensics and Security*.
4. Lei Zhang, **Meng Yang**, Xiangchu Feng, Yi Ma, and David Zhang, "Collaborative Representation based Classification for Face Recognition," *IEEE Transactions on Image Processing* (under review).
5. **Meng Yang**, Lei Zhang, Simon C. K. Shiu, and David Zhang, "Robust Kernel Representation for Face Recognition with Statistical Local Feature," *IEEE Transactions on Neural Networks and Learning Systems* (under review).
6. Zhizhao Feng, **Meng Yang**, Lei Zhang, Yan Liu and David Zhang, "Joint Discriminative Dimensionality Reduction and Dictionary Learning for Face Recognition", *Pattern Recognition* (under major revision).
7. **Meng Yang**, Lei Zhang, Xiangchu Feng, and David Zhang, "Fisher Discrimination Dictionary Learning for Sparse Representation," under preparation.

Acknowledgements

Foremost, I would like to thank my advisor, Dr. Lei Zhang, who have shared his expertise with me during the three years and have been of invaluable help. His scientific vision, passion, high criterion, and detailed comments have been a major source of inspiration and motivation for my work. Time after time, he showed me how to express ideas clearly and write academic papers deeply. His extraordinary enthusiasm for research impacts me greatly. What I have learnt and experienced during the time I spent in his laboratory will benefit me much in the future.

I would also like to thank Prof. David Zhang (my co-supervisor) and Prof. Jane You for their support and providing BRC a great research environment.

I would like to express my gratitude to many people who have directly or indirectly contributed to the work in this thesis: Prof. Xiangchu Feng, Prof. Shiguang Shan, Prof. Jian Yang, Prof. Qinghua Hu, Prof. Chunhou Zheng, Dr. Zhenhua Guo, Dr. Guangming Lu, Dr. Weisheng Dong, Dr. Qijun Zhao, Dr. Lin Zhang, Jin Xie, Dr. Po Peng, Kaihua Zhang, Zhizhao Feng, Pengfei Zhu, Nan Luo, Xiaofeng Qu, Xingzheng Wang, Dr. Dongmin Guo, Feng Liu, Yahui Liu, Jinghua Wang, Dr. Qin Li, Dr. Guojun Liu, Dr. Yafeng Li, Xuande Zhang, Dr. Min Zhang, Wufeng Xue, Shenlong Wang, and all other members of the PolyU BRC group that I cannot enumerate here. I thank them for interesting discussions and suggestions on my work. In addition, we share the pain and pleasure of the PhD studies or working at The Hong Kong Polytechnic University.

My deepest gratitude goes to my family for their mental support and encouragement, my parents Ziming Yang and Xiuying Qian, my younger brother Jin Yang, and my beautiful and considerate girlfriend Ying Zhu. They will not read this thesis, but without their love and support I can’t complete this journey.

Table of Content

List of Figures

List of Tables

Chapter 1. Introduction

1.1 Face Recognition

1.1.1 Introduction of face recognition

1.1.1.1 Face recognition system

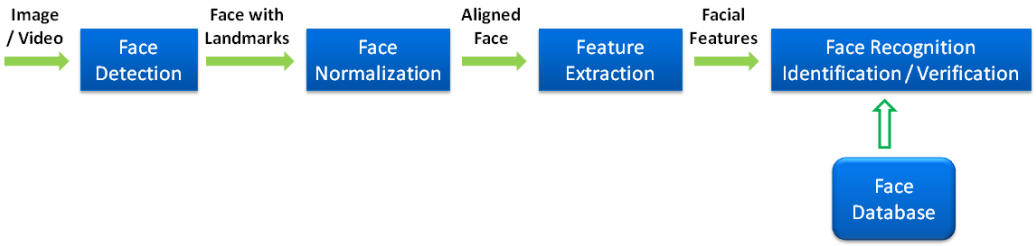


Figure 1.1: Flow chart of a generic face recognition system.

A facial recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video source. The pipeline of a generic face recognition system is shown in Fig. 1.1. Given still or video images, first face detection is applied to detect the facial region with landmarks (e.g., locations of eyes' corners or centers). Based on the landmarks, the facial region is cropped, aligned and normalized, and then facial features are extracted from the aligned face. Finally, by matching with the enrolled template face images, face recognition (either identification or verification) can be done [1-2]. In face identification, the input to the system is an unknown face, and the system reports back the testing image's identity from a database with known individuals. In face verification, the input is an unknown face image with a claimed identity, and the system needs to decide whether the individual is who he/she claims to be [1, 3].

1.1.1.2 Utilities of face recognition

Face is a reliable biometrics trait, and thus face recognition is a biometrics recognition problem. A biometrics system is essentially a pattern recognition system which uses data collected from human subjects as patterns. It extracts a feature set from the acquired biometrics data, and compares this feature set against the template sets in the database [4]. Apart from face, there are many kinds of biometrics traits, e.g., DNA [5], iris [6-10], retina [11-12], ear [13-14], fingerprint [15-19], finger-knuckle-print [20-21], hand geometry [22-23], voice [24-25], signature [26-27], palmprint [28-35], and gait [36], etc. Images of these representative biometrics traits are shown in Fig. 1.2.

Figure 1.2: Some representative biometric identifiers.

Face recognition (FR) has a wide range of applications, e.g., information security, access control, surveillance, smart cards, law enforcement, human computer interaction, and entertainment [1-3]. In comparison with other biometrics traits, FR has some clear advantages, as summarized in Table 1.1. Firstly, FR has a wider range of applications, such as surveillance in public place, multimedia management, human computer interaction, entertainment, etc. Secondly, face is the least intrusive biometric identifier,

which does not even require the cooperation of participants, and does not carry any sanitation risks. FR also has very low cost, only needing a camera and processor. Finally, FR is probably the most common biometric characteristic used by humans to make personal recognition, and it has been attracting significant attentions from the field of computer vision, image/video processing, pattern recognition, biometrics, biological vision and machine learning.

Table 1.1: Advantages of FR over other biometrics traits.

Keywords	Detailed information
More applications	surveillance in public place (e.g., FaceIt-Hist [37]), witness face reconstruction [38], social networks (e.g., photo tagging in Facebook [39]), multimedia management (e.g., face based search), human computer interaction (e.g., face synthesis and animation [40]), criminal justice systems (mug-shot/booking systems, post-event analysis, forensics), and entertainment (e.g., video game), etc.
Least intrusive	capturing face images at a distance, contactless, can be free of cooperation of participants, and free of any sanitation risks.
Low cost	cost-effective equipment, e.g., a camera and a processor.
More data sources	ubiquitous images and videos, e.g., passports, internet photos, ATM camera, and video surveillance.
Human habits	the most common biometric characteristic used by humans to make a personal recognition
Academic research merits	widely studied in computer vision, image/video processing, pattern recognition, biometrics, biological vision and machine learning, etc.

1.1.1.3 Challenges of face recognition

The wide range of practical applications (e.g., commercial and law enforcement) and high merits of academic research make face recognition a hot research topic in the past decades. After 40 years of research, the face recognition systems in well controlled indoor environment have reached a certain level of maturity. However, for the more challenging applications in uncontrolled and outdoor environments, current systems are still far away from matured.

The main challenges of FR to be tackled include:

1) *Variations from the face itself—pose, expression and aging:* Human face is a 3D object, which will have very different appearance under different viewpoints. Facial expression will also make the same face show very different looking. In addition, the aging-related phenomena, such as speckles, wrinkles, and changes in shape of face primitives (e.g., sagged cheeks, eyes or mouth) would result in large variations of face images from the same subject, degrading the face recognition performance.
2) *Variations from external factors—illumination and various occlusion and disguise:* various illumination conditions and shading could be generated by different lighting sources. Face occlusion (e.g., pixel corruption and block occlusion) and disguise (e.g., hair, makeup, sunglasses, and scarf) can often appear in face images. Both illumination and occlusion would greatly change the appearance of face images, resulting in small inter-class variation and large intra-class variation.
3) *Huge number of face classes—human being:* In the world there are billions of persons, while many people look very similar. Such a huge number of face classes and the similarity of face patterns make highly accurate FR very difficult.
4) *Small-sample-size problem—high-dimensional data but few or single training sample:* the learnt classifier is easy to perform very well in the training data but poorly in testing data.

1.1.2 A brief review of face recognition technologies

The earliest work on FR can be traced back at least to the 1950s in psychology [41] and to 1960s in engineering [42]. The first PhD thesis for FR was done by T. Kanade in 1973 [43]. In the primary stage, there are two types of techniques applied to the FR with frontal views. The first type, also the first approach toward an automated recognition of faces, is based on the computation of a set of ***geometrical features*** (e.g., relative position and other parameters of distinctive features such as eyes, mouth, nose, and chin) from the picture of a face [43-44]. The second class of techniques is based on ***template matching***. Some FR

methods based on template matching were reviewed by [44].

Recent years have witnessed the rapid development and deployment of face recognition and modeling systems. This is evidenced by the emergence of face recognition conferences such as the *International Conference on Audio- and Video-Based Biometric Person Authentication* (AVBPA) since 1997 and the *IEEE International Conference on Automatic Face and Gesture Recognition* (AFGR) since 1995. And many systematic empirical evaluations of face recognition techniques, including the FERET [45], FRVT [46-48], and XM2VTS [49] protocols, and commercial systems are available [1, 4, 146].

Several FR surveys have reviewed the many FR technologies [1-4, 50-53], which could be categorized by different ways, e.g., 2D/3D FR [51], still-image/video based FR, and homogeneous/heterogeneous FR [4, 53-55], etc. Here, we briefly review the main FR technologies from the following four aspects: subspace-based face representation (e.g., principal component analysis (PCA) [57], kernel PCA (KPCA) [70]), model-based face representation (e.g., active appearance model (AAM) [75-77]), texture-based face representation (e.g., local binary pattern (LBP) [90]) and representation-based classification (e.g., nearest subspace (NS) [97, 99-101] and sparse representation based classification (SRC) [102]) (see Fig. 1.3).

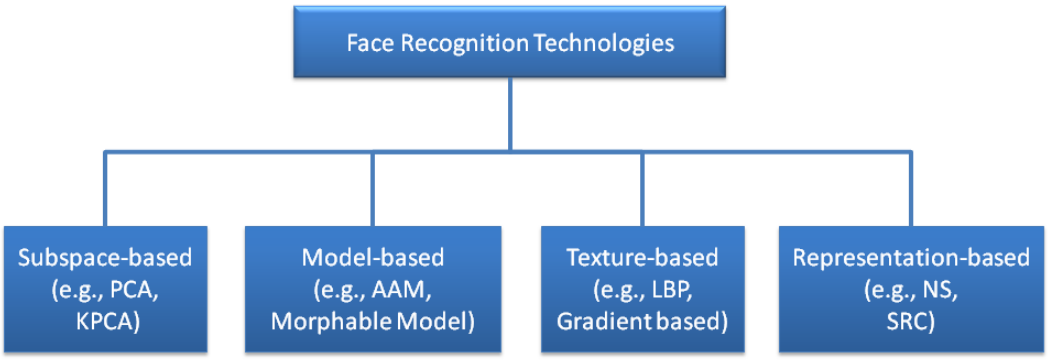


Figure 1.3: Main face recognition technologies.

1.1.2.1 Subspace-based face representation

Subspace-based FR is a kind of appearance-based holistic approach. Although facial images have a high dimensionality, they usually lie in a manifold with intrinsically low dimension. By mapping the high-dimensional face image into a lower dimensional subspace, a compact representation of the face pattern can be generated.

In the communities of face recognition and computer vision, there is a growing interest to apply algebraic and statistical tools to extract and analyze the underlying manifold of face images [56]. The classical Eigenfaces and Fisherfaces [57-58] algorithms are the most representative subspace methods based on principal component analysis (PCA) and linear discriminant analysis (LDA), respectively. Compared to Eigenfaces [57], Fisherfaces [58] introduces discrimination information by maximizing the ratio of between-class scatter to within-class scatter. PCA [57] minimizes the sample covariance (second-order dependence) of the face data, while the higher-order dependencies can be minimized by the independent component analysis (ICA) [59-60]. All PCA [57], LDA [58] and ICA [59-60] could be called linear subspace-based technologies for that they assume a linear principal manifold. Besides, as extensions of Eigenfaces, probabilistic Eigenspaces [61-62], Bayesian algorithm using probabilistic subspace [63-64], and Tensorfaces [65-66] are also representative linear subspace-based FR approaches.

Although linear subspace-based FR (e.g., Eigenfaces [57] and Fisherfaces [58]) have been widely used, they consider only the global scatter of training samples and fail to reveal the essential data structures nonlinearly embedded in high dimensional space. To overcome these limitations, the nonlinear manifold learning methods were proposed [67-68]. The nonlinear principal manifold is a nonlinear (curved) lower-dimensional surface, often referred to as principal curves [69]. The principal curves pass through the middle of the data while minimizing the total distance between the data points and their projections on the manifold. Kernel extension of linear subspace-based FR is another category of nonlinear subspace-based approaches. By using a kernel, the original linear operation can be done in a reproducing kernel Hilbert space with a non-linear mapping.

The representative kernel subspace approaches are kernel PCA [70], kernel Fisherfaces (KLDA) [71], etc.

1.1.2.2 Model-based face representation

This type of FR approaches attempt to build a 2D or 3D face model to facilitate the recognition of face images in the presence of pose, expression, illumination and age variations. A representative model-based FR is elastic bunch graph matching (EBGM) [78], which represents a face as a labeled image graph with each node being a fiducial point. By extracting Gabor coefficients on each node and using fiducial points, EBGM could deal with FR with pose variation, even the tilted or occluded face. More recent model-based technologies could be divided into face alignment models, morphable face models and face aging models [2].

Face alignment models: A statistical approach is adopted to learn the way in which the face shape and texture vary across a large and representative training set of face images. The typical face alignment models are Active Shape Model (ASM) [72-74] and Active Appearance Model (AAM) [75-77]. In ASM, a set of feature points (e.g., eyes, nose, mouth and eye-brows) are annotated to define correspondence across the training set, from which the statistical model of shape variation could be learnt and new face shapes could be synthesized. In order to utilize the strong discrimination of face texture, AAM applies similar techniques used in ASM to build a statistical model of face appearance. AAM combines the power of statistical models of face shape and appearance, and achieves state-of-the-art performance in dealing with face alignment.

Morphable face models: morphable face modeling aims to build a synthesis framework, which is able to generate all possible face images. In order to be applicable to all input face images, morphable model of faces consists of a 3D shape and appearance model plus an imaging model, which could not only enable the accurate modeling of any illumination and pose but also well separate these variations from the rest (e.g., identity and expression). Given an input image, the morphable model searches for its best

parameters to make the generated image as similar as possible to the input image. Representative morphable model based FR are expression-invariant 3D FR [79], pose and illumination invariant FR [80-81], etc.

Face aging models: face aging is an important cause that degrades the performance of FR systems. Building 2D or 3D face aging model that can compensate for the aging process is one of the most successful solutions to age-invariant FR. Recently, Ramanathan *et al.* [82] modeled the face shape growth up to age 18; Lanitis *et al*. [83] built an aging function in terms of PCA coefficients of shape and texture; and age invariant FR by learning aging pattern from PCA coefficients of separated 3D shape and 2D texture was proposed in [84].

1.1.2.3 Texture-based face representation

In texture-based FR technologies, robust local features are extracted. Two important texture-based technologies are FR based on gradient information and FR based on local statistical features (i.e., statistical information of micro patterns).

FR based on gradient information: The face feature of the raw intensity values is quite sensitive to the changes of ambient lighting. It has been found that image gradient information is insensitive and robust to different illuminations (e.g., uncontrolled and natural lighting). Based on the reflectance model of face image, Gradientfaces [85] based on the ratio of y-gradient to x-gradient was proposed for illumination invariant FR. Recently, it was found that the distribution of gradient orientation differences of two pixel-wise dissimilar image approximates a uniform distribution [86]. Based on the measurement of cosine-based correlation of gradient orientations, PCA-based [86], sparse representation-based [87], and subspace learning-based [88] FR approaches were proposed.

FR based on local statistical features: Unlike many appearance-based FR methods, which are either holistic feature based (e.g., Eigenface [57] and Fisherface [58]) or local feature based (e.g., Gabor feature based classification [89]), the adoption of local binary

pattern (LBP) in FR [90] triggers the use of local statistical features (LSF) in the FR field. LSF-FR methods consist of two main phases: statistical histogram feature extraction and feature selection (e.g., weighting the histogram feature extracted in different blocks). Histogram feature extraction could be further divided into three steps: feature map generation (e.g., original image, Gabor feature), pattern map coding (e.g., LBP) and sub-region histogram computing. Almost all the LSF-FR methods [90-95] have similar procedures of sub-region histogram computing (i.e., extracting the statistical information of pattern feature in each sub-region), which shows certain robustness to local deformations (e.g., variations of pose, expression, and occlusion) of face images. However, the different schemes of feature map generation and pattern map coding leads to different LSF-FR methods.

The well-known LBP operator [90] directly uses the image intensity values to encode the image local pattern features. In order to overcome the sensitiveness of pixel intensity value to the image variations (e.g., illumination), Zhang *et al.* [91] proposed to extract directional Gabor magnitude features at multiple scales, and then apply LBP to the Gabor magnitude feature maps for robust LSF. The study of Gabor phase based LSF-FR methods were conducted in [92-94]. Zhang *et al.* [92] adopted multi-scale Gabor phase to take the place of Gabor magnitude in [91], and the global and local variations of real part and imagery part of complex Gabor filtering coefficients were encoded in [93]. Recently, Xie *et al.* [94] utilized XOR (exclusive or) operator to encode the local variation of Gabor phase, and then fused Gabor-magnitude local pattern and Gabor-phase local pattern. This scheme achieves very promising FR results.

1.1.2.4 Representation based technologies

Unlike the subspace-based and texture-based technologies, which focus on extracting effective facial feature, and the model-based technologies, which focus on building a generative model to pose and expression, representation based technologies pay more attention to how to classify the face image or extracted features based on linear

combination of training images or their features. Given a sufficient expressive training set, the recognition of a testing face image is accomplished by checking which class could result in the minimal distance between the testing sample and it. Based on the labels of training samples to represent the testing image, representation based technologies could be categorized as within-class representation methods and across-class representation methods.

Within-class representation: all the training samples used to represent the testing image belong to the same class. The most popular classifier for FR may be the nearest neighbor (NN) classifier due to its simplicity and efficiency. In order to overcome NN's limitation that only one training sample is used to represent the testing face image with representation coefficient as 1, Li and Lu proposed the nearest feature line (NFL) classifier [96], which uses two training samples for each class to represent the testing face. Chien and Wu [97] then proposed the nearest feature plane (NSP) classifier, which uses three independent samples to represent the testing image. Later on, classifiers using more training samples for face representation were proposed, such as the local subspace classifier (LSC) [98] and the nearest subspace (NS) classifiers [97, 99-101], which represent the testing sample by all the training samples of each class.

Across-class representation: a testing face image could be represented collaboratively across different classes. The representative work is sparse presentation based classification [102], which parsimoniously selects training samples from all the classes to represent the testing sample. With across-class representation, constraints on the representation coefficients, e.g., structural sparse constraint [103], nonnegative sparse constraint [104], and joint sparse representation [105], can be imposed in order for robust face recognition performance.

1.1.3 New trends of face recognition

As the rapid development of computer science, imaging technologies, and especially

network, there are several new trends of FR.

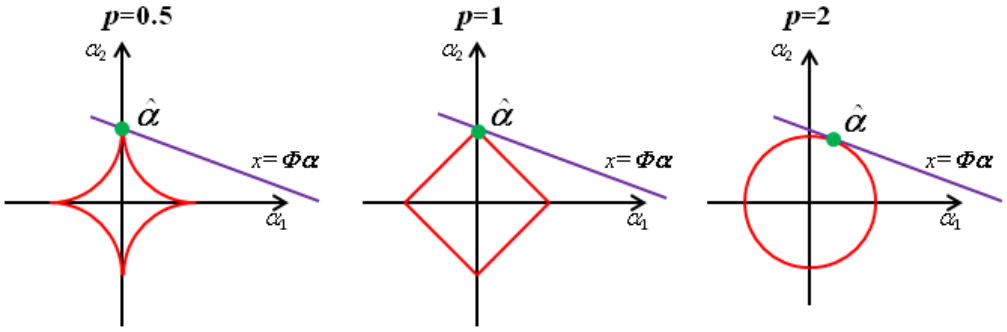
1) *Web-based uncontrolled FR:* Recent years have witnessed the rapid development of FR using face images collected from internet under uncontrolled environment. The representative databases include LFW [106] and PubFig [107]. Some corresponding FR methods for such databases can be found in [107-110].
2) *High-resolution still/video FR:* As indicated by FRGC [111], high resolution images are one of three main contenders for improving FR performance. Current FR systems mainly rely on low-resolution still images and videos, which lead to the lose of important information contained in the microscopic traits [112]. More discrimination information for accurate FR would be contained in the high-resolution images/videos, which consist of facial images with about 250 pixels between the centers of the eyes in average [111].
3) *Heterogeneous FR:* As defined in [4], "Heterogeneous face recognition refers to matching face images across different image formats that have different image formation characteristics." Typical examples are matching 2D images to 3D models, visible light images to infrared images [54], photo to sketches [55]. There are great needs for heterogeneous FR in practical applications. For instance, in the application of law enforcement where the photo image of a suspect is not available, a sketch drawing based on the recollection of an eyewitness is one of the best substitutes [55]. Therefore, automatic face photo-sketch synthesis and recognition becomes important. However, heterogeneous FR has additional difficulty for that heterogeneity can increase the intra-class variability.

1.2 Sparse Representation

Natural images can be generally coded by structural primitives, e.g., edges and line segments [113], and these primitives are qualitatively similar in form to simple cell

receptive fields [114-115]. In [114, 116], Olshausen *et al.* proposed a sparse coding strategy of image representation, i.e., representing a natural image using a small number of basis functions chosen out of an over-complete code set. Partially due to the progress of l_0-norm and l_1-norm minimization techniques [117-126], in recent years, such a sparse coding or sparse representation strategy has been widely studied to solve inverse problems, and researchers have achieved a big success in various applications, include compressive sensing [127], morphological component analysis [128], image restoration [129-136], and super-resolution [137-138], etc.

Suppose that $\boldsymbol{x}\in\Re^n$ is the target signal to be coded, and $\boldsymbol{\Phi}=[\boldsymbol{\phi}_1,\ldots,\boldsymbol{\phi}_m]$ is a given dictionary of atoms $\boldsymbol{\phi}_i$ (i.e., code set), the sparse coding of $\boldsymbol{x}$ over $\boldsymbol{\Phi}$ is to find a sparse representation vector $\boldsymbol{\alpha}$ (i.e., most of the coefficients in $\boldsymbol{\alpha}$ are close to zero) such that $\boldsymbol{x}\approx\boldsymbol{\Phi\alpha}$ [119]. The general regularized coding to solve $\boldsymbol{\alpha}$ is $\hat{\boldsymbol{\alpha}}=\arg\min_{\boldsymbol{\alpha}}\|\boldsymbol{\alpha}\|_{l_p}$ s.t. $\boldsymbol{x}\approx\boldsymbol{\Phi\alpha}$, where $\|\cdot\|_{l_p}$ is the l_p norm. When $p\le 1$, the regularized coding problem requires $\boldsymbol{\alpha}$ to be sparse. A simple example of regularized coding is shown in Fig. 1.4, where $\boldsymbol{\alpha}=[\alpha_1;\alpha_2]$, x is a scalar, the line shows the solution of $x=\boldsymbol{\Phi\alpha}$, and the red graph denotes the set of $\boldsymbol{\alpha}$ with equal l_p-norm. Fig. 1.4(a) shows the non-convex sparse coding with p=0.5, Fig. 1.4(b) shows a convex sparse coding with p=1, and a non-sparse coding with p=2 is shown in Fig. 1.4(c).



(a) (b) (c)

Figure 1.4: Regularized coding in two dimensional space. (a) non-convex sparse coding with p=0.5, (b) convex sparse coding with p=1, and (c) convex non-sparse coding with p=2.

If the sparsity is measured as the l_0-norm of $\boldsymbol{\alpha}$, which counts the number of non-zero coefficients in $\boldsymbol{\alpha}$, the sparse coding problem becomes $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0$ s.t. $\|\boldsymbol{x}-\boldsymbol{\Phi\alpha}\|_2^2 \le \varepsilon$, where ε is a small number to tolerate the noise. However, the combinatorial l_0-norm minimization is an NP-hard problem, and greedy pursuit algorithms [119, 125] are often used to solve it. Hence, the l_1-norm minimization, as the closest convex function to l_0-norm minimization, is widely employed for sparse coding: $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\|\boldsymbol{x}-\boldsymbol{\Phi\alpha}\|_2^2 \le \varepsilon$ [120-121]. It was also shown that l_0-norm and l_1-norm minimizations are equivalent if the solution is sufficiently sparse [121, 140]. In statistics, the l_1-norm minimization technique is equivalent to the so-called LASSO problem [117-118], which was developed in the context of variable selection. The objective function of LASSO is $\min_{\boldsymbol{\alpha}} \|\boldsymbol{x}-\boldsymbol{\Phi\alpha}\|_2^2$ s.t. $\|\boldsymbol{\alpha}\|_1 \le \delta$, where δ is a constant. More algorithms for sparse coding solutions can be found in a recent review [124, 126].

One successfully application of sparse coding is signal/image restoration. In the inverse problems such as signal/image restoration, often we have a degraded observation $\boldsymbol{y}=\boldsymbol{H}\boldsymbol{x}+\boldsymbol{\upsilon}$ of the target signal $\boldsymbol{x}$, where $\boldsymbol{H}$ is the degrading operator and $\boldsymbol{\upsilon}$ is additive noise. Based on the different forms of $\boldsymbol{H}$, the problems can be denoising, deblurring, super-resolution, inpainting and compressive sensing, etc [127-139]. In general, the sparse solution to the inverse problem can be obtained by $\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y}-\boldsymbol{H\Phi\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right\}$, where λ is a regularization parameter controlling the sparsity of the solution. Many algorithms have been proposed to solve the above minimization problems, such as the *surrogate* [132], *proximal* [141] and the TwIST [133] algorithms. Another impressive application of sparse representation is compressive (or compressed) sensing [127]. Instead of first sampling enough number of samples (suppose N samples) of a compressible signal $\boldsymbol{x}$ and then compressing it, the compressive sensing theory supports sampling directly M ($M<<N$)

linear measurements of $\boldsymbol{x}$ with a random sensing matrix $\boldsymbol{\Psi}$, i.e., $\boldsymbol{y}=\boldsymbol{\Psi x}$. It is shown [140, 142-143] that if the sensing matrix $\boldsymbol{\Psi}$ satisfies the *restricted isometry property*, the original signal $\boldsymbol{x}$ can be effectively reconstructed. Recently, sparse representation has been successfully used in pattern classification, including face recognition [102-105, 144-145, 147].

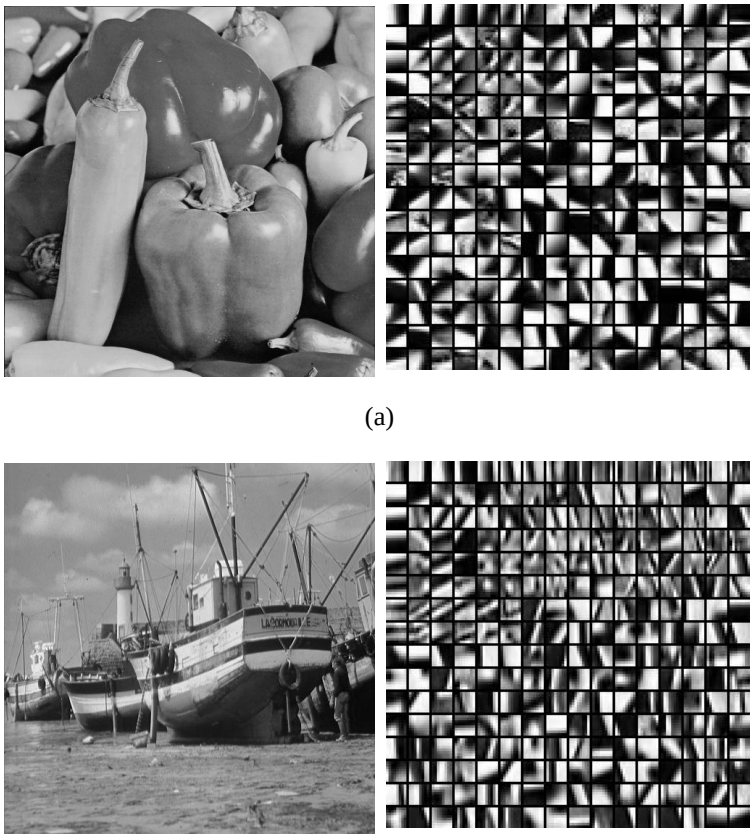
Beyond the l_0-norm or l_1-norm sparse coding, in [104, 164] the representation coefficients are also required non-negative to pursue more physical meanings. More interestingly, the mixed l_1/l_p norm is widely used in group sparse coding [165-167] for meaningful feature representation and used in joint sparse coding [105, 168-169] for multi-task representation. More recently, structured sparsity was also proposed to enforce specific patterns of non-zero coefficients in different applications [103, 170-172].

1.3 Dictionary Learning

In sparse and redundant representation, the target signal is described as a linear combination of a few atoms from an over-completed dictionary. Therefore, the choice of the dictionary which can sparsely represent the signals is crucial for the success of this model.

Generally speaking, there are two ways to build a proper dictionary [148]: 1) building a dictionary with off-the-shelf bases designed via a mathematical model, or 2) learning a dictionary from a training set. The dictionary built by the first way could be seen as an analytic dictionary, and the examples of such dictionaries include wavelets, curvelets, contourlets, and bandelets, etc [148]. Although the analytically designed dictionary has several advantages, such as free of training samples and universality to various types of signals, it may not be effective enough in specific tasks. Learning dictionaries from example signals/images can be a much more effective approach to dictionary design. The advancement of sparse representation theory and algorithms has strongly influenced the

development of dictionary learning, and most dictionary learning methods train dictionary atoms in the scheme of sparse representation regularized by l_0 or l_1 sparsity constraint. Compared to analytic dictionaries, the main advantage of learnt dictionaries lies in that they lead to state-of-the-art results in many practical applications, such as image restoration [130, 135-136], image denoising [129, 134], image super-resolution [150], image compression [151], unsupervised clustering [152], edge detection and image interpretation [153], and pattern classification [145, 155-159], etc.



(a)

(b)

Figure 1.5: Original image and learnt dictionary from its patches by K-SVD [129]. (a) Example of "pepper", and (b) example of "boat".

Two classical dictionary learning methods are method of optimal directions (MOD) [160-161] and K-SVD algorithm [129], both of which could be formulated as

$\min_{\boldsymbol{D},\boldsymbol{\Gamma}} \|\boldsymbol{A}-\boldsymbol{D}\boldsymbol{\Gamma}\|_F^2$ s.t. $\|\boldsymbol{\gamma}_i\|_0 \le \varepsilon,\ \forall i$, where $\boldsymbol{A}$ is the training sample matrix, $\boldsymbol{\Gamma}$ is the coding vector matrix, $\boldsymbol{\gamma}_i$ is the coding vector of the i^{th} training sample, and $\boldsymbol{D}$ is the dictionary to be learnt. In the training phase, both of them alternate sparse coding and dictionary update steps. In the step of sparse coding (i.e., $\min_{\boldsymbol{\Gamma}} \|\boldsymbol{A}-\boldsymbol{D}\boldsymbol{\Gamma}\|_F^2$ s.t. $\|\boldsymbol{\gamma}_i\|_0 \le \varepsilon,\ \forall i$), each signal vector in $\boldsymbol{A}$ is individually and sparsely represented on the fixed dictionary $\boldsymbol{D}$. In the step of dictionary updating (i.e., $\min_{\boldsymbol{D}} \|\boldsymbol{A}-\boldsymbol{D}\boldsymbol{\Gamma}\|_F^2$), MOD solves the entire dictionary by an analytic solution using a matrix inversion, while K-SVD updates the dictionary atom-by-atom in a simple and efficient process. Fig.1.5 shows the dictionary learning results from natural image patches by K-SVD. It can be seen that the learnt dictionaries are similar for different images. Inspired by KSVD, other dictionary learning methods such as coupled dictionary learning [150] have also been proposed for video restoration and image super-resolution. In addition, online dictionary learning [163] has also been developed. A recent review of dictionary learning can be found in [148].

More recently, dictionary learning is applied to pattern classification, and it achieves state-of-the-art results in the applications of digit recognition [156], texture classification [156, 162], face recognition [155, 157, 159], clustering [152, 158] and image classification [145], etc. Overall, dictionary learning for image and face classification has a great potential that deserves deeper investigation.

1.4 Outline of the Thesis and Our Contributions

Although sparse representation and dictionary learning have received significant attentions and achieved state-of-the-art results in many fields, such as image processing, compressive sensing, etc., the research of sparse representation and dictionary learning for pattern classification (e.g., face recognition) is still in its infancy. Many problems need to be

further investigated, including the role of sparsity in pattern classification, effective feature extraction, robust measurement of representation residuals and dictionary choosing, etc. In this thesis, we explore these problems in details.

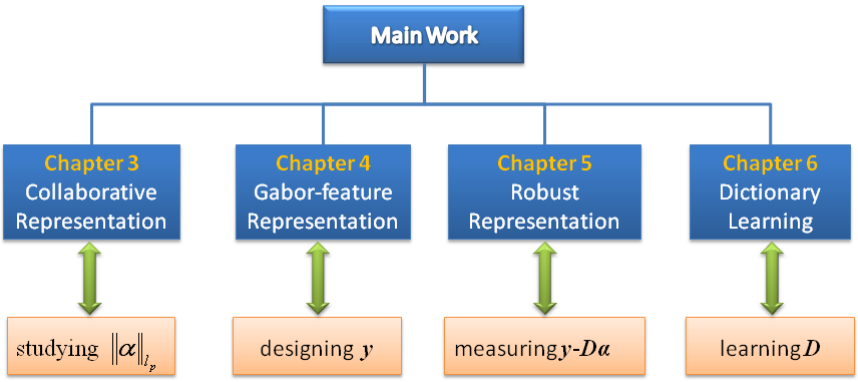


Figure 1.6: Main work of this thesis.

Chapter 2 introduces the related works on representation based classification. Chapter 3 discusses the mechanism of sparse representation based classification (SRC) and the role of l_1-norm sparsity, and then presents the collaborative representation based classification (CRC) scheme. Chapter 4 studies the use of local features to improve the FR performance, and presents the Gabor feature based representation and classification scheme. Chapter 5 focuses on how to robustly measure representation residuals, and presents a regularized robust coding model for face recognition. Then in Chapter 6 a discriminative dictionary learning method based on Fisher discrimination criterion for pattern classification is presented. Finally the summary of this thesis are given in Chapter 7, with some open problems and future work also discussed. As shown in Fig. 1.6, the main works are presented in Chapters 3, 4, 5 and 6, which have close relations. Chapter 3 presents a fundamental work of the Chapters 4, 5 and 6. Chapters 4 and 5 present two different ways to increase the robustness to facial occlusions, while the proposed method in Chapter 5 is more robust to various occlusions. Different from directly using the training samples as the dictionary in Chapters 3, 4 and 5, Chapter 6 improves the former chapters by learning a

discriminative dictionary. A single face recognition system which combines Chapters 3, 4/5, and 6 could be built.

This thesis brings several contributions to the fields of pattern classification, machine learning and computer vision. As shown in Fig. 1.6, considering an unconstrained representation model: $\min_{\boldsymbol{\alpha}} \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2+\lambda\|\boldsymbol{\alpha}\|_{l_p}$, we discussed the following key issues in this thesis: the collaborative representation of $\boldsymbol{y}$ using the training samples from all classes and the role of l_p norm regularization on $\boldsymbol{\alpha}$ (Chapter 3), Gabor-feature representation of $\boldsymbol{y}$ and the Gabor occlusion dictionary learning (Chapter 4), robust measure of the fidelity term $\boldsymbol{y}$-$\boldsymbol{D}\boldsymbol{\alpha}$ (Chapter 5), and how to learn a discriminative dictionary $\boldsymbol{D}$ (Chapter 6). The main contributions of this thesis are summarized as follows.

- We propose a novel representation model for face classification, namely collaborative representation based classification (CRC) in Chapter 3. By illustrating how sparse representation based classification (SRC) works, we show that it is the collaborative representation (i.e., representing the testing image collaboratively by samples from all the classes) but not the l_1-norm sparse representation that makes SRC effective for face recognition, and the proposed l_2-norm regularized CRC could achieve similar/better performance to/than the l_1-norm sparse representation. The SRC is a special case of collaborative representation based classification (CRC), which has various instantiations by applying different norms to the coding residual and coding coefficient. We verify the face recognition accuracy and efficiency of the CRC scheme with different instantiations.
- We show in Chapter 4 how to exploit Gabor feature based representation for more accurate face classification performance with lower computational burden. The use of Gabor features not only increases the face discrimination power, but also allows us to compute a compact Gabor occlusion dictionary, on which the coding coefficients could be regularized by l_2-norm. The proposed approach demonstrates the high

effectiveness and efficiency on representative face databases with variations of lighting, expression, pose and occlusion.

- We propose in Chapter 5 a regularized robust representation (RRC) scheme, which can effectively model face images with various outliers. The RRC is a maximum a posterior solution of the coding problem with the assumption that the coding residual and the coding coefficient are respectively independent and identically distributed. An efficient algorithm for solving RRC is also presented. Extensive experiments on representative face databases demonstrate that the RRC is much more effective and efficient than state-of-the-art sparse representation based methods in dealing with face occlusion, corruption, lighting and expression changes, etc.
- We show in Chapter 6 how to learn a discriminative structured dictionary (i.e., dictionary atoms have correspondence to the class label) for pattern classification. The learnt dictionary could not only make the class-specific representation residual discriminative (i.e., the class-specific dictionary could well represent the samples with the same label but have low representation ability to samples of other classes), but also make the coding coefficients have small within-class scatter but big between-class scatter. With a new classification scheme combining the discriminative information in the representation residual and sparse coding coefficients, the proposed FDDL leads to (or approaches to) state-of-the-art results for several pattern recognition problems such as face recognition, digit recognition, gender classification and object categorization.

Chapter 2. Related Works on Representation based Classification

In this chapter, we first review the main methods for representation based classification, and then discuss the sparse representation based classification in detail; at last, the dictionary learning methods for pattern classification are reviewed.

2.1 Overview of Representation based Classification

A fundamental problem in pattern classification is to correctly determine the label of a testing sample by using the labeled training samples. Usually there are two steps in representation based classification. The first step is to represent the testing sample on the training samples and output the representation coefficient, denoted by α. Then the classification is accomplished by checking which class could result in the minimal distance between the testing sample and its representation on the training samples.

There are a number of works on estimating the coefficient α for classification tasks, such as face recognition (FR). Based on the labels of training samples used to represent the testing image, representation based classification technologies could be categorized as within-class representation based ones and across-class representation based ones. Nearest neighbor and nearest subspace classifiers [96-101, 173-178] are typical within-class representation based classification methods, while sparse representation based classification [102] is the most representative method of across-class representation. The representation based classification could also be extended to kernel space, such as [147]. Here we focus on the linear representation models. In the following we briefly review the major works.

2.1.1 Within-class representation based classification

Denote by $\boldsymbol{A}_i = [\boldsymbol{s}_{i,1}, \boldsymbol{s}_{i,2}, ..., \boldsymbol{s}_{i,n_i}] \in \Re^{m \times n_i}$ the set of training samples of the i^{th} object class, where $\boldsymbol{s}_{i,j}$, j=1,2,…,n_i, is an m-dimensional vector stretched by the j^{th} sample of the i^{th} class, and n_i the number of training samples in the i^{th} class. Given a sufficient training dataset $\boldsymbol{A}_c$, intuitively a testing sample $\boldsymbol{y} \in \Re^m$ from the c^{th} class could be well approximated by the linear combination of the samples within $\boldsymbol{A}_c$, i.e., $\boldsymbol{y} \approx \sum_{j=1}^{n_c} \alpha_{c,j} \boldsymbol{s}_{c,j} = \boldsymbol{A}_c \boldsymbol{\alpha}_c$, where $\boldsymbol{\alpha}_c = [\alpha_{c,1}, \alpha_{c,2}, ..., \alpha_{c,n_c}]^T \in \Re^{n_c}$ is the coding vector. Therefore the best representation for the testing sample $\boldsymbol{y}$ could be sought class by class, and the class label of the testing sample is determined by checking which class could have the best representation accuracy.

2.1.1.1 Nearest neighbor (NN) classifier

The nearest neighbor (NN) classifier finds the nearest training sample to testing sample $\boldsymbol{y}$, and uses the nearest neighbor as the best representation in class c:

$$r_c^{NN} = \min\left\{ \left\| \boldsymbol{e}_j \right\|_2 : j \in \{1, 2, \cdots, n_c\} \right\} \tag{2-1}$$

where $\boldsymbol{e}_j = \boldsymbol{y} - \boldsymbol{s}_{c,j}$ is the representation residual vector, and r_c^{NN} is the minimal distance between $\boldsymbol{y}$ and class c. The classification is made via:

$$\text{identity}(\boldsymbol{y}) = \arg\min_c \left\{ r_c^{NN} \right\} \tag{2-2}$$

2.1.1.2 Nearest feature line

The simplest extension of NN is the nearest feature line (NFL) proposed by Li and Lu [96], which searches along an optimal pair of training samples the best representation of $\boldsymbol{y}$:

$$r_c^{NFL} = \min\left\{ \left\| \boldsymbol{e}_{j,k} \right\|_2 : j, k \in \{1, 2, \cdots, n_c\}, j \neq k, \alpha_1 \in \Re^1 \right\} \tag{2-3}$$

where $\boldsymbol{e}_{j,k} = \boldsymbol{y} - \alpha_1 \boldsymbol{s}_{c,j} - (1 - \alpha_1) \boldsymbol{s}_{c,k}$ is the representation residual vector, and r_c^{NFL} is the minimal distance between $\boldsymbol{y}$ and class c. The classification is the same as in Eq. (2-2).

2.1.1.3 Nearest feature plane

Following NN and NFL, Chien and Wu [97] extended the geometric concepts of point (i.e., only one sample used in NN) and line (i.e., only two samples used in NFL) to plane. In nearest feature plane (NFP), the distance between $\boldsymbol{y}$ and i^{th} class is defined as

$$r_c^{NFP} = \min\left\{\left\|\boldsymbol{e}_{j,k,g}\right\|_2 : j,k,g \in \{1,2,\cdots,n_c\}, j \neq k, j \neq g, k \neq g\right\} \tag{2-4}$$

where $\boldsymbol{e}_{j,k,g} = \boldsymbol{y} - \boldsymbol{p}_{j,k,g}^c$ and $\boldsymbol{p}_{j,k,g}^c$ is the projection of $\boldsymbol{y}$ on plane $F_{j,k,g}^c$, which is spanned by three linear independent feature points (e.g., $\boldsymbol{s}_{c,j}$, $\boldsymbol{s}_{c,k}$ and $\boldsymbol{s}_{c,g}$) of class c. r_c^{NFP} is then taken as the minimal distance between $\boldsymbol{y}$ and class c, and the classification is performed as that in Eq. (2-2).

2.1.1.4 Nearest subspace

Chien and Wu [97] further generalized the geometrical concept from plane to space (i.e., the subspace spanned by all the independent prototype features associated with class i), and Li [100] also proposed a nearest subspace classifier, which linearly combines all the training samples of a certain class to represent $\boldsymbol{y}$:

$$r_c^{LC} = \min\left\{\left\|\boldsymbol{y} - \boldsymbol{A}_c\boldsymbol{\alpha}_c\right\|_2 : \boldsymbol{\alpha}_c \in \Re^{n_c \times 1}\right\} \tag{2-5}$$

which is called nearest linear combination (NLC) or nearest constrained linear combination (NCLC) if constrained by $\sum_{j=1}^{n_c} \boldsymbol{\alpha}_c(j) = 1$. Eq. (2-5) is a K-Local hyperplane algorithm [173] from a different geometric point of view.

Eq. (2-5) could be computed with an analytical solution of $\boldsymbol{\alpha}_c$ [98, 101]:

$$\boldsymbol{\alpha}_c = \left(\boldsymbol{A}_c^T\boldsymbol{A}_c\right)^{-1}\boldsymbol{A}_c^T\boldsymbol{y} \tag{2-6}$$

With the solved distance, the classification can be performed as that in Eq. (2-2). The downsampled images were used in the local regression classification (LRC) [101] algorithm, which is a kind of nearest subspace method and achieves good performance in FR.

The application of nearest subspace classifiers to FR has empirical and analytical

supports, especially under variable lighting [174-176]. It was found that under certain assumptions, face images under all possible lighting conditions form an illumination cone [176]. Furthermore, Basri and Jacobs [175], and Ramamoorthi and Hanrahan [177-178] have shown that the illumination cone can be accurately approximated by a 9-dimensional linear subspace. Lee *et al.* [99] has also shown how to acquire linear subspaces for FR under variable lighting. However, in practical FR, sufficient training samples for each class can not be guaranteed, resulting in the performance degradation of nearest subspace based classification methods.

2.1.2 Across-class representation based classification

Although the nearest subspace classifier adopts more samples than the NN, NFL and NFP classifiers in testing sample representation, the cross-class combination of training samples is not allowed. Recently, it has been shown that across-class representation is very helpful to deal with the small-sample-size problem and avoid the overfitting problem in FR [102, 179-180].

Suppose that we have K object classes, and let $\boldsymbol{A}=[\boldsymbol{A}_1, \boldsymbol{A}_2,\ldots, \boldsymbol{A}_K]$ be the concatenation of the n training samples from all the K classes, where $n=n_1+n_2+\ldots+n_K$, then a testing sample $\boldsymbol{y}$ with $\boldsymbol{y} \approx \boldsymbol{A}_c\boldsymbol{\alpha}_c$ could also be well reconstructed by the linear combination of all the training samples:

$$\boldsymbol{y} \approx \boldsymbol{A}\boldsymbol{\alpha} \tag{2-7}$$

where one solution to the representation coefficient $\boldsymbol{\alpha}$ could be $\boldsymbol{\alpha}=[\boldsymbol{\alpha}_1; \ldots, \boldsymbol{\alpha}_c; \ldots; \boldsymbol{\alpha}_K] = [\mathbf{0}; \ldots; \boldsymbol{\alpha}_c; \ldots; \mathbf{0}]$. In other words, the significant elements in $\boldsymbol{\alpha}$ could identify the identity of testing sample $\boldsymbol{y}$. The main problem of across-class representation based classification is to estimate the representation coefficient $\boldsymbol{\alpha}$ which should have enough discrimination.

In Eq. (2-7), the best representation of the testing sample $\boldsymbol{y}$ using all classes can be solved by

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_2 \text{ s.t. } R(\boldsymbol{\alpha}) \tag{2-8}$$

where $R(\boldsymbol{\alpha})$ is some regularization imposed on $\boldsymbol{\alpha}$. Because the matrix $\boldsymbol{A} \in \Re^{m \times n}$, which is composed of all the training samples from all classes, is often over-complete (e.g., $n>m$) or very redundant in the problem of face representation, some regularization on the representation coefficient $\boldsymbol{\alpha}$ is needed to make the representation stable. It can be seen that the across-class representation can be interpreted as collaborative representation across classes, as well as competitive representation between classes in view of the following classification stage. Collaborative representation is helpful to overcome the small-sample-size problem and it can make the testing sample be well represented. Meanwhile, it would benefit the final classification because if one class contributes more in the representation, other classes will contribute less. In the face representation problem, "collaboration" and "competition" are the two sides of the same coin.

The across-class representation is used in the recently proposed sparse representation based classification (SRC) [102]. SRC mainly involves two steps: a testing face image is firstly coded over all the training images with l_1-norm sparsity imposed on the coding vector; then the classification is performed by checking which class has the least coding residual. SRC has led to state-of-the-art performance in robust face recognition (e.g., with random pixel corruption, random block occlusion and various disguise). The success of SRC triggers the research of pattern classification by sparse representation and dictionary learning [103-105, 147, 157, 162, 181-185, 196], which are reviewed in the following sections.

2.2 Sparse Representation based Classification

The great success of sparse representation in compressive sensing [127] and image processing [129-131, 134-136, 150-151, 186] triggers the research on sparse representation based pattern classification. The basic idea is to code the testing sample

over an over-complete dictionary with sparsity constraint, and then do classification based on the coding vector. It is believed that the sparsity constraint will make the coding vector more discriminative so that the classification accuracy can be improved. Under such a philosophy, Huang and Aviyente [144] sparsely coded a signal over a set of predefined redundant bases and took the coding vector as features for classification. Rodriguez and Sapiro [145] learnt a discriminative dictionary under the sparse representation framework and used it to code the image for classification.

One pioneer work that applies sparse representation to robust FR is the sparse representation based classification (SRC) scheme [102]. In SRC for FR without occlusion, a testing sample $\boldsymbol{y}_0$ is sparsely coded on the training dataset $\boldsymbol{A}$ via l_1-minimization

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \text{ s.t. } \boldsymbol{y}_0 = \boldsymbol{A}\boldsymbol{\alpha} \tag{2-9}$$

Then classification is made by

$$\text{identity}(\boldsymbol{y}_0) = \arg\min_i \{r_i\} \tag{2-10}$$

where $r_i = \|\boldsymbol{y}_0 - \boldsymbol{A}_i\delta_i(\hat{\boldsymbol{\alpha}})\|_2$, $\hat{\boldsymbol{\alpha}} = [\delta_1(\hat{\boldsymbol{\alpha}});\cdots;\delta_i(\hat{\boldsymbol{\alpha}});\cdots;\delta_K(\hat{\boldsymbol{\alpha}})]$, and $\delta_i(\cdot):\Re^n \to \Re^{n_i}$ is the characteristic function which selects the coefficients associated to the i^{th} class from the original coding coefficients.

In SRC for FR with occlusion or corruption, the testing sample $\boldsymbol{y}$ is sparsely coded as

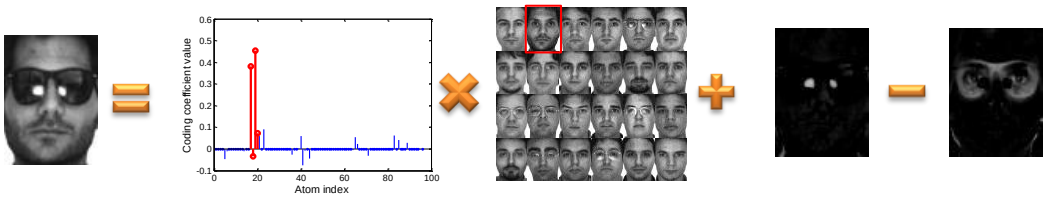
$$[\hat{\boldsymbol{\alpha}}_0;\hat{\boldsymbol{\alpha}}_e] = \arg\min_{\boldsymbol{\alpha}_0,\boldsymbol{\alpha}_e} \|[\boldsymbol{\alpha}_0;\boldsymbol{\alpha}_e]\|_1 \text{ s.t. } \boldsymbol{y} = [\boldsymbol{A},\boldsymbol{A}_e][\boldsymbol{\alpha}_0;\boldsymbol{\alpha}_e] \tag{2-11}$$

where $\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{e}_0 = \boldsymbol{A}\boldsymbol{\alpha}_0 + \boldsymbol{A}_e\boldsymbol{\alpha}_e$, $\boldsymbol{y}_0$ is the clean face image, and $\boldsymbol{e}_0$ is the corruption error. $\boldsymbol{y}_0$ and $\boldsymbol{e}_0$ are expected to have sparse representations over the training sample dictionary $\boldsymbol{A}$ and the occlusion dictionary $\boldsymbol{A}_e \in \Re^{m\times n_e}$, respectively. The corruption dictionary $\boldsymbol{A}_e$ is set as an identity matrix $\boldsymbol{I}$ in SRC [102]. Then classification is made by Eq. (2-10) with r_i computed by

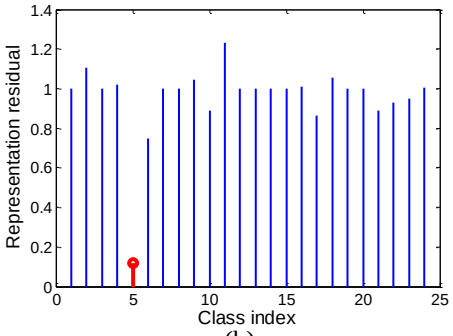
$$r_i = \|\boldsymbol{y} - \boldsymbol{A}_i\delta_i(\boldsymbol{\alpha}) - \boldsymbol{A}_e\boldsymbol{\alpha}_e\|_2 \tag{2-12}$$

Fig. 2.1 gives an example of FR with sunglass by using SRC. Fig. 2.1(a) shows the sparse representation (solved by Eq. (2-11)) of the input face image, where red

coefficients correspond to training images of the correct individual (in the red box), and the right two images are the estimated corruption. Fig. 2.1(b) illustrates the representation residuals (i.e., r_i in Eq. (2-12)) associated to each class, which clearly show that the correct class has the lowest reconstruction error.



(a)



(b)

Figure 2.1: An example of sparse representation based classification [102]. (a) Sparse representation of the face image. (b) Representation residuals associated to each class for classification.

The success of SRC boosts the research of sparsity based FR, and many works have been consequently reported. For instance, sparse representation has been extended to kernel space [147]; various improved sparsity constraints, e.g., structural sparse constraint [103], nonnegative sparse constraint [104, 154], and joint sparse representation [105], have been proposed for classification. Sparse representation based robust FR with continuous occlusion [187] and misalignment [188-189] has also been studied. Besides, l_1-graph has been proposed for image classification [182] and subspace learning [183]. In [190], sparse coding was combined with linear spatial pyramid matching for image classification. In [191], sparse representation with low-rank decomposition was proposed

to align a batch of linearly correlated image with gross corruption. In addition, dictionary learning methods [157, 162, 184-185, 196] were also developed to enhance SRC based pattern classification, which will be discussed in next section.

Though sparse representation has shown very promising performance in classification tasks, especially in robust FR, there are still some significant concerns:

1) *The role of l_1-sparsity and the working mechanism of SRC are not fully revealed yet*. Many following works of SRC aim to improve the l_1-regularization term on coding vector $\boldsymbol{\alpha}$. For example, Liu *et al.* [104] added a nonnegative constraint to $\boldsymbol{\alpha}$; Gao *et al.* [192] introduced a Laplacian term of $\boldsymbol{\alpha}$ in sparse coding; Yuan and Yan [105] used joint and group sparse representation to code the multiple types of image features; and Elhamifar and Vidal [103] used structured sparse representation for robust classification. All these works stress the role of l_1-sparsity of $\boldsymbol{\alpha}$ in classification. However, the role of collaborative representation in SRC, i.e., using the training samples from all classes to represent the testing sample $\boldsymbol{y}$, is rather ignored. In the recent work [193, 238], the role of sparsity in classification has been questioned. This issue will be discussed in detail in Chapter 3, and the related works of collaborative representation based classification have been published in [179-180].
2) *Holistic and local features*. SRC adopts holistic features (e.g., Eigenfaces, Randomfaces, raw intensity value) and this makes the size of occlusion dictionary $\boldsymbol{A}_e$ very big and makes SRC computationally expensive. This issue is not fully solved by the following sparsity based FR methods [103, 147, 157, 162, 184-185, 187-189]. For instance, only holistic features are considered in [103, 157, 162, 184-185, 187-189, 197], and FR with occlusion is ignored in [147, 188], while the problem of large occlusion dictionary is not addressed in [157, 162, 184-185, 197]. To solve this issue, we will present in Chapter 4 the algorithm of Gabor feature based representation and Gabor occlusion dictionary learning [181, 258].

3) *Robustness to outliers*. In Eq. (2-9) and Eq. (2-11), the coding residual $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}$ is measured by the l_2- and l_1-norm, respectively, which actually assumes that the coding residual $\boldsymbol{e}$ follows Gaussian or Laplacian distribution. In practice, however, such an assumption may not hold well, especially when occlusions, corruptions and expression variations occur in the testing face images. Few works have been reported to solve this problem. In Chapter 5, we will propose a regularized robust coding model [194-195] by seeking for a maximum a posterior solution of the coding problem, which can effectively deal with FR with various kinds of outliers.

2.3 Dictionary Learning based Classification

As reviewed in Section 1.3, the choice of dictionary that sparsely represents the signals is crucial for the success of sparse representation model, and learning dictionary from training data by enforcing sparsity constraint on coding coefficients has lead to state-of-the-art results in many practical applications, such as image restoration [130, 135-136], image denoising [129, 134], image super-resolution [150], image compression [151], unsupervised clustering [152], etc. Inspired by the great success of dictionary learning in the above fields, dictionary learning based classification has also been paid much attention to and promising performance has been achieved [145, 155-159, 162, 196].

The very basic model of dictionary learning could be written as

$$\min_{\boldsymbol{D},\boldsymbol{X}} \left\|\boldsymbol{A}-\boldsymbol{D}\boldsymbol{X}\right\|_F^2 \quad \text{s.t.} \left\|\boldsymbol{x}_i\right\|_0 \le \varepsilon \ \text{ or } \left\|\boldsymbol{x}_i\right\|_1 \le \varepsilon \ \ \forall i \tag{2-13}$$

where $\boldsymbol{A}$ is the training dataset, $\boldsymbol{x}_i$ represents a column of coding coefficient matrix $\boldsymbol{X}$, and $\boldsymbol{D}$ is the dictionary to be learnt. Usually, each column $\boldsymbol{d}_j$ of the dictionary is required to satisfy $\left\|\boldsymbol{d}_j\right\|_2^2 \le 1$. The representative dictionary learning methods, such as KSVD [129] and MOD [160-161], learn the dictionary by solving Eq. (2-13). However, Eq. (2-13) may

not be suitable for classification tasks because it can only ensure that the learnt dictionary $\boldsymbol{D}$ could faithfully represent the training samples $\boldsymbol{A}$.

In the task of classification, usually additional priors on the dictionary and/or the representation coefficients are introduced in the phase of dictionary learning [144, 152, 155-159, 162, 165, 173-174, 184, 196]. The general dictionary learning model for classification tasks could be represented as

$$\min_{\boldsymbol{D},\boldsymbol{X}} \left\|\boldsymbol{A}-\boldsymbol{D}\boldsymbol{X}\right\|_F^2 \text{ s.t.} \begin{cases} \text{Prior}\left(\boldsymbol{X}\right) \\ \text{Prior}\left(\boldsymbol{D}\right) \\ \left\|\boldsymbol{x}_i\right\|_0 \le \varepsilon \ \text{ or } \left\|\boldsymbol{x}_i\right\|_1 \le \varepsilon \ \ \forall i \end{cases} \tag{2-14}$$

where the constraint Prior($\boldsymbol{X}$) could introduce discrimination information to the representation coefficients (e.g., training a coefficient based classifier [155-157, 159, 184, 196]), and the constraint Prior($\boldsymbol{D}$) could make the class-specific representation residuals discriminative [152, 158, 162]. According to the types of constraints (e.g., on $\boldsymbol{X}$ or on $\boldsymbol{D}$), we review the dictionary learning methods for classification by two categories, dictionary learning with additional Prior($\boldsymbol{X}$) and dictionary learning with additional Prior($\boldsymbol{D}$).

2.3.1 Dictionary learning with Prior($\boldsymbol{D}$)

In this case, the atoms of learnt dictionary have labels to define the correspondence to different classes. Usually the atoms of such class-specific dictionary should be able to well reconstruct the training samples of the same class, but have poor representation ability to other classes. Based on KSVD, Mairal *et al.* [162] added a reconstruction-error constraint in the dictionary learning model to gain certain discrimination ability, and applied the learnt dictionary to texture segmentation and scene analysis. Sprechmann *et al.* [152] optimized a set of dictionaries, one for each cluster, with which the signals can be well clustered. Later on, Ramirez *et al.* [158] added an incoherence promoting term to the model in [152], encouraging dictionaries associated with different classes to be as independent as possible.

2.3.2 Dictionary learning with Prior(X)

In this category, the learnt dictionary is shared by all classes. In other words, all the training samples could be well reconstructed by the atoms of the shared dictionary. Rodriguez *et al.* [145] proposed to learn dictionary with discriminative sparse coefficients by using an orthogonal-matching-pursuit like method. Dictionary learning by training a linear classifier was also proposed for digit recognition, texture classification [156], and object categorization and FR [155]. Based on [155] and KSVD [129], Zhang *et al.* [157] proposed an algorithm called discriminative KSVD (DKSVD) for FR, followed by the so-called Label-Consistent K-SVD [184]. Local feature based dictionary was also learnt via back-projection in [159] to represent local features. Recently, beyond l_0- or l_1-norm sparsity, nonnegative [164], group [165] and structured [170] sparsity constraints were proposed in different applications to enforce specific patterns of non-zero coefficients.

Although the first category of dictionary learning methods enforces discrimination to the class-specific representation residuals, it does not enforce discrimination to the representation coefficients in training the dictionary and doing classification. For the second category of dictionary learning methods, the shared dictionary loses the correspondence between the dictionary atoms and the class labels, and hence performing classification based on the reconstruction error associated with each class is not allowed. To exploit the discriminative information in both representation residual and coefficients, we will propose a dictionary learning method based on Fisher discrimination criterion, which will be presented in Chapter 6.

2.4 Summary

In this chapter, some related works of this thesis, including representation based classification, sparse representation and dictionary learning based classification, were reviewed. Sparse representation and dictionary learning have led to state-of-the-art results

in many applications such as image reconstruction. However, the study of sparse representation and dictionary learning based classification is still in its infancy. This thesis aims to investigate the various issues in this problem and advance the research of sparse representation and dictionary learning in computer vision. In the following Chapters 3~5, we will focus on sparse representation based face classification, and dictionary learning will be discussed in Chapter 6.

Chapter 3. Collaborative Representation based Classification for Face Recognition

3.1 Introduction

Inspired by the findings of sparsity in human visual perception [114-115], sparse representation or sparse coding has been successfully used in many applications, including compressive sensing [127], morphological component analysis [186], image restoration [129-131, 135], and super-resolution [137-138], etc. The great success of sparse representation in image reconstruction triggers the research on sparse representation based pattern classification [102, 144, 145, 147, 157, 162, 182-185, 188, 189], as described in Section 2.2.

The pioneer work by using sparse representation was reported by Wright *et al.* [102] for face recognition (FR). Denote by $\boldsymbol{A}_i \in \Re^{m\times n_i}$ the set of training samples from class i (each column of $\boldsymbol{A}_i$ is a sample). Suppose that we have K classes of subjects, and let $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K]$. The so-called sparse representation based classification (SRC) [102] scheme mainly involves two steps. Firstly, a testing face image $\boldsymbol{y}\in\Re^m$ is coded on $\boldsymbol{A}$ by $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right\}$ for standard SRC (S-SRC) or $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha},\boldsymbol{e}} \left\{ \|\boldsymbol{y} - [\boldsymbol{A}, \boldsymbol{I}][\boldsymbol{\alpha}; \boldsymbol{e}]\|_2^2 + \lambda \|[\boldsymbol{\alpha}; \boldsymbol{e}]\|_1 \right\}$ for robust SRC (R-SRC). Secondly the classification is performed by checking which class has the least coding residual. It is not difficult to see that the latter coding model is basically equivalent to $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_1 + \lambda \|\boldsymbol{\alpha}\|_1 \right\}$ [202] (this equivalence could be derived by denoting $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\alpha}$); that is, the coding residual is also characterized by l_1-norm to achieve robustness to outliers.

SRC (including S-SRC and R-SRC) shows very interesting and powerful FR performance, and it inspires many following works, e.g., kernel sparse representation [147], l_1-graph for image classification [182-183], misalignment robust FR [188-189], image aligning [191] and dictionary learning [157, 162, 185]. Beyond that, many fast algorithms have been proposed to speed up the l_1-minimization process [126, 198-204]. As reviewed in [126], there are five representative fast l_1-norm minimization approaches, namely, Gradient Projection, Homotopy, Iterative Shrinkage-Thresholding, Proximal Gradient, and Augmented Lagrange Multiplier (ALM). It was indicated in [126] that for noisy data, the first order l_1-minimization techniques (e.g., SpaRSA [200], FISTA [201], and ALM [202]) are more efficient, while in the application of FR, Homotopy [203], ALM and l_1_ls [204] are better for their good accuracy and fast speed.

Though SRC shows interesting results in FR and has been widely studied in the community, its working mechanism is not fully revealed yet. The role of l_1-sparsity is often emphasized in face classification, and many works aim to improve the l_1-regularization term on coding vector $\boldsymbol{\alpha}$ [103-105, 192]. However, recently Berkes *et al.* [193] argued that there is no clear evidence for active sparsification in the visual cortex. More importantly, the role of collaborative representation in SRC, i.e., using the training samples from all classes to represent the testing sample $\boldsymbol{y}$, is rather ignored.

As reviewed in Section 2.1, the SRC classifier has a close relationship to the nearest classifiers, including the nearest neighbor (NN), nearest feature line (NFL) [100], nearest feature plane (NSP) [97], and nearest subspace (NS) [96-99, 101] classifiers. The NN, NFL and NSP classifiers use one, two and three training samples, respectively, to represent the testing image for classification, while the NS classifiers represent the testing sample by all the training samples of each class. Like these nearest classifiers, SRC also represents $\boldsymbol{y}$ as the linear combination of training samples; however, one critical difference between SRC and these classifiers is that SRC collaboratively represents $\boldsymbol{y}$ by training samples from all classes, while the nearest classifiers represent $\boldsymbol{y}$ by each individual class. The use of all classes to collaboratively represent $\boldsymbol{y}$ alleviates much the small-sample-size

problem in FR, especially when number of training samples per class is relatively small.

In this Chapter, we discuss the collaborative representation nature of SRC, and present a more general model, namely collaborative representation based classification (CRC), for FR. By using either l_1-norm or l_2-norm to characterize the coding vector $\boldsymbol{\alpha}$ and the coding residual $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\alpha}$, we can have different instantiations of CRC, while S-SRC and R-SRC are special cases of CRC. More specifically, the l_1- or l_2-norm characterization of $\boldsymbol{e}$ is related to the robustness of CRC to outlier facial pixels, while the l_1- or l_2-norm characterization of $\boldsymbol{\alpha}$ is related to the discrimination of employed facial feature $\boldsymbol{y}$. When the face image is not occluded/corrupted, l_2-norm is good enough to model $\boldsymbol{e}$; when the face image is occluded/corrupted, l_1-norm is more robust to model $\boldsymbol{e}$. The discrimination of facial feature $\boldsymbol{y}$ is often related to its dimensionality. If the dimensionality and hence the discrimination of $\boldsymbol{y}$ is high, the coding coefficients $\boldsymbol{\alpha}$ will be naturally sparse and concentrate on the samples whose class label is the same as $\boldsymbol{y}$, no matter l_1- or l_2-norm is used to regularize $\boldsymbol{\alpha}$. When the dimensionality of $\boldsymbol{y}$ is low, often the discrimination power of $\boldsymbol{y}$ will be reduced, and thus the distribution of $\boldsymbol{\alpha}$ will be less sparse, and some big coefficients can be generated and assigned to the samples whose class labels are different from $\boldsymbol{y}$. In this case, the l_1-norm regularization on $\boldsymbol{\alpha}$ will enforce $\boldsymbol{\alpha}$ to be sparse, and consequently enhance its discrimination power. Considering that the l_1-regularization on $\boldsymbol{\alpha}$ will make the computational cost high, and usually the facial feature $\boldsymbol{y}$ can have a high enough dimensionality, a good instantiation of CRC is that we use l_2-norm (for non-occluded and non-corrupted faces) or l_1-norm (for occluded/corrupted faces) to measure $\boldsymbol{e}$, and use l_2-norm to regularize $\boldsymbol{\alpha}$. Such a modeling can lead to not only high FR rate but also low computational complexity.

3.2 The Role of Sparsity in Representation based FR

There are two key points in SRC [102]: (i) the coding vector $\boldsymbol{\alpha}$ is enforced to be sparse (regularized by the l_1-norm), and (ii) the coding of testing sample $\boldsymbol{y}$ is performed over the whole dataset $\boldsymbol{A}$ instead of each subset $\boldsymbol{A}_i$. It was claimed in [102] that the sparsest (or the most compact) representation of $\boldsymbol{y}$ over $\boldsymbol{A}$ is naturally discriminative and thus can indicate the identity of $\boldsymbol{y}$. As we explained in the Section 3.1, the SRC classifier is a generalization and significant extension of classical nearest classifiers such as NN and NS by representing $\boldsymbol{y}$ collaboratively across classes. But there are some issues not very clear yet: why the sparsity constraint on $\boldsymbol{\alpha}$ makes the representation more discriminative, and must we impose l_1-norm sparsity on $\boldsymbol{\alpha}$ to this end?

Denote by $\boldsymbol{\Phi} \in \Re^{m \times n}$ a dictionary of bases (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 and complete, however, often we need to use many atoms from $\boldsymbol{\Phi}$ to faithfully represent $\boldsymbol{x}$. If we want to use less number of atoms to represent $\boldsymbol{x}$, we must relax the orthogonality requirement on $\boldsymbol{\Phi}$. In other words, we should allow more atoms to be involved in $\boldsymbol{\Phi}$ so that we have more choices to represent $\boldsymbol{x}$ using the atoms in $\boldsymbol{\Phi}$, leading to an over-complete and redundant dictionary $\boldsymbol{\Phi}$ but a sparser representation of signal $\boldsymbol{x}$. The recent great success of sparse representation in image restoration [129-131, 135] validates that a redundant dictionary can have more powerful capability to represent and reconstruct the signal.

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 code of much lower dimensionality. Let's take the set of training samples of class i, i.e., $\boldsymbol{A}_i$, as the dictionary for this class. In practice the atoms (i.e., the training samples) of $\boldsymbol{A}_i$ will be correlated. Assume that we have enough training samples for each class and all the face images of class i can be faithfully represented by $\boldsymbol{A}_i$, then $\boldsymbol{A}_i$ can be viewed as a redundant

dictionary[1] because of the correlation of training samples of class *i*. Therefore, we can conclude that a testing sample $\boldsymbol{y}$ of class *i* can be sparsely represented by dictionary $\boldsymbol{A}_i$.

Another important fact in FR is that human faces are all somewhat similar, and some subjects may have very similar face images. That is, dictionary $\boldsymbol{A}_i$ of class *i* and dictionary $\boldsymbol{A}_j$ of class *j* are not incoherent; instead, they can be highly correlated. Using the NS classifier, for a testing sample $\boldsymbol{y}$ from class *i*, we can find (by least square method) a coding vector $\boldsymbol{\alpha}_i$ such that $\boldsymbol{\alpha}_i = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{A}_i\boldsymbol{\alpha}\|_2^2$. Let $\boldsymbol{r}_i=\boldsymbol{y}-\boldsymbol{A}_i\boldsymbol{\alpha}_i$. Similarly, if we represent $\boldsymbol{y}$ by class *j*, there is $\boldsymbol{\alpha}_j = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{A}_j\boldsymbol{\alpha}\|_2^2$ and we let $\boldsymbol{r}_j=\boldsymbol{y}-\boldsymbol{A}_j\boldsymbol{\alpha}_j$. For the convenience of discussion, we assume that $\boldsymbol{A}_i$ and $\boldsymbol{A}_j$ have the same number of atoms, i.e., $\boldsymbol{A}_i, \boldsymbol{A}_j \in \Re^{m\times n}$. Let $\boldsymbol{A}_j = \boldsymbol{A}_i + \boldsymbol{\Delta}$. When $\boldsymbol{A}_i$ and $\boldsymbol{A}_j$ are very similar, $\boldsymbol{\Delta}$ can be very small such that $\xi = \frac{\|\boldsymbol{\Delta}\|_F}{\|\boldsymbol{A}_i\|_F} \le \frac{\sigma_n(\boldsymbol{A}_i)}{\sigma_1(\boldsymbol{A}_i)}$, where $\sigma_1(\boldsymbol{A}_i)$ and $\sigma_n(\boldsymbol{A}_i)$ are the largest and smallest eigenvalues of $\boldsymbol{A}_i$, respectively. Then we can have the following relationship between $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ (Theorem 5.3.1, page 242, [205]):

$$\frac{\|\boldsymbol{r}_j - \boldsymbol{r}_i\|_2}{\|\boldsymbol{y}\|_2} \le \xi\left(1+\kappa_2(\boldsymbol{A}_i)\right)\min\{1, m-n\} + O\left(\xi^2\right) \tag{3-1}$$

where $\kappa_2(\boldsymbol{A}_i)$ is the l_2-norm conditional number of $\boldsymbol{A}_i$. From Eq. (3-1), we can see that if $\boldsymbol{\Delta}$ is very small, the distance between $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ will also be very small. This makes the classification very unstable because some small disturbance can make $\|\boldsymbol{r}_j\|_2<\|\boldsymbol{r}_i\|_2$, leading to a wrong classification.

The above problem can be much alleviated by regularization, for example, enforcing some sparsity on $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_j$. The reason is very intuitive. Take the l_0-norm sparsity regularization as an example, 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 $\boldsymbol{A}_i$ to represent $\boldsymbol{y}$ with a good accuracy. In contrast, we may need more samples, e.g., 8 or 9 samples, in $\boldsymbol{A}_j$ to represent $\boldsymbol{y}$ with nearly the same

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

representation accuracy. With the sparsity constraint or other regularizer, the representation error of $\boldsymbol{y}$ by $\boldsymbol{A}_i$ will be visibly lower than that by $\boldsymbol{A}_j$, making the classification of $\boldsymbol{y}$ easier. Here let's consider three regularizers: the sparse regularizers by l_0-norm and l_1-norm, and the l_2-norm least square regularizer.

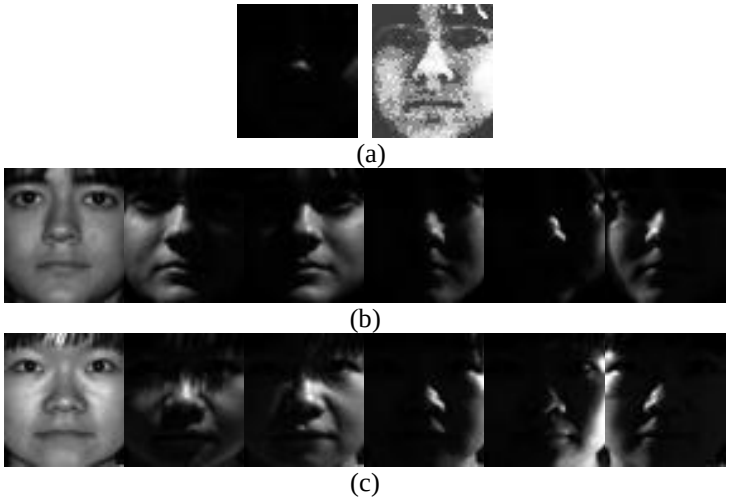


Figure 3.1: An example of class-specific face representation. (a) The testing face image (left: original image; right: the one after histogram equalization for better visualization); (b) some training samples from the class of the testing image; (c) some training samples from another class.

By l_p-regularization, p = 0, 1, or 2, the representation of $\boldsymbol{y}$ by dictionary $\boldsymbol{\Phi}$ can be formulated as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 \text{ s.t. } \|\boldsymbol{\alpha}\|_{l_p} \le \varepsilon \tag{3-2}$$

where ε is a positive number. Let $r = \|\boldsymbol{y} - \boldsymbol{\Phi}\hat{\boldsymbol{\alpha}}\|_2$. We could plot the curves of "r vs. ε" for two similar classes to illustrate how regularization improves discrimination. Fig. 3.1(a) shows a testing face image of class 32 in the Extended Yale B database [99, 206]. Some training samples of this class are shown in Fig. 3.1(b), while some training samples of class 5, which is similar to class 32, are shown in Fig. 3.1(c). We use the training samples of the two classes as dictionaries to represent, respectively, the testing sample in Fig. 3.1(a) by using Eq. (3-2). The "r vs. ε" curves for p = 0, 1, and 2 are drawn in Fig. 3.2 (a), Fig.

3.2(b) and Fig. 3.3(c), respectively. For the l_0-norm regularization, we used the Orthogonal Matching Pursuit (OMP) algorithm [125] to solve Eq. (3-2); for the l_1-norm regularization, we used the l_1_ls algorithm [204]; while for l_2-norm regularization, the regularized least square can be used to get an analytical solution to Eq. (3-2).

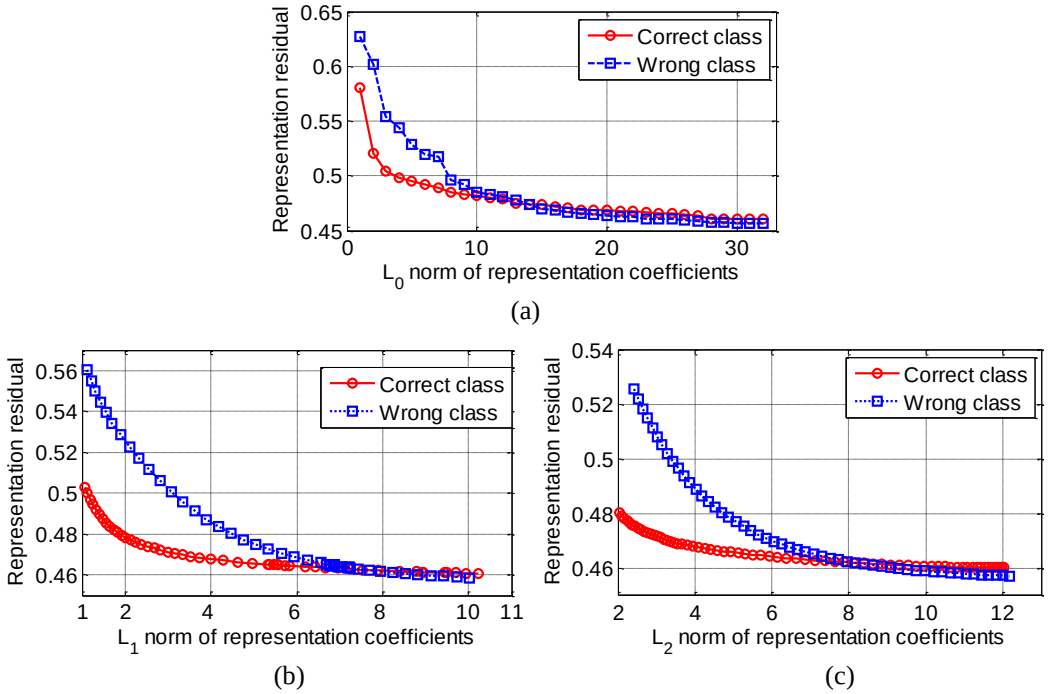


Figure 3.2: The curve of representation residual versus the l_p-norm of the representation coefficients. (a) p=0, (b) p=1, and (c) p=2.

From Fig. 3.2(a), one can see that when using only a few training samples (e.g., less than 3 samples) to represent the testing sample, both the two classes have big representation error. In practice, the system will consider this sample as an imposter and directly reject it. When more and more training samples are involved, the representation residual r decreases. However, the ability of r to discriminate the two classes will also reduce if too many samples (e.g., more than 10 samples) are used to represent the testing sample. This is because the two classes are similar so that the dictionary of one class can represent the samples of another class if enough training samples are available (i.e., the dictionary is nearly over-complete). With these observations, we can conclude that a

testing sample should be classified to the class which could faithfully represent it using less number of samples, and the l_0-norm sparse regularization on $\boldsymbol{\alpha}$ can do this job. In other words, the sparsity can improve the discrimination of representation based classification.

Now the question is: can the weaker l_1-norm sparsity, and even the non-sparse l_2-norm regularization, do a similar job? Fig. 3.2(b) and Fig. 3.2(c) give the answer, where one can see that l_1-norm and l_2-norm regularizations also work well in improving the discrimination of representation based classification. When ε is big (e.g., ε>8; here the feature vector is normalized to have unit l_2-norm), which means that the regularization is lose, both the two classes have very low reconstruction residual e, making the classification very unstable. By setting a smaller ε, the l_1-norm or l_2-norm regularized solution of $\boldsymbol{\alpha}$ will result in discriminative reconstruction residual, by which the testing sample can be correctly classified. From this example, it can be concluded that the role of l_0-norm or l_1-norm sparsity on $\boldsymbol{\alpha}$ is basically to regularize the solution, while the non-sparse l_2-norm regularization can play a similar role to sparse l_1-norm regularization in face classification by regularing the solution.

Remark (regularized nearest subspace, RNS): The above observations and discussions imply a regularized nearest subspace (RNS) scheme for FR when the number of training samples of each class is big. In this case, we can represent the testing sample $\boldsymbol{y}$ class by class, and classify it based on the representation residual and regularization strength. Since l_0-norm minimization is combinatorial and NP-hard, it is more practical to use l_1-norm or l_2-norm to regularize the representation coefficients. Using the Lagrangian formulation, we have the objective function of RNS-L_p as:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_{l_p} \right\} \tag{3-3}$$

where p = 1 or 2 and λ is a positive constant. For each class $\boldsymbol{A}_i$, we could obtain its representation vector $\hat{\boldsymbol{\alpha}}_i$ of $\boldsymbol{y}$ by taking $\boldsymbol{\Phi}$ as $\boldsymbol{A}_i$ in Eq. (3-3). Denote by

$r_i = \|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i\|_2^2 + \lambda\|\hat{\boldsymbol{\alpha}}_i\|_{l_p}$ the sum of representation residual and regularization strength. We can then classify $\boldsymbol{y}$ by $\text{identity}(\boldsymbol{y}) = \arg\min_i \{r_i\}$.

3.3 The Collaborative Representation based Classification (CRC)

In our discussion in Section 3.2, we assumed that there are enough training samples each class so that each dictionary $\boldsymbol{A}_i$ is redundant. Unfortunately, FR is a typical small-sample-size problem, and $\boldsymbol{A}_i$ is under-complete in general. If we use $\boldsymbol{A}_i$ to represent $\boldsymbol{y}$, the representation residual r_i can be big, even when $\boldsymbol{y}$ is from class i. Consequently, the classification based on r_i will be unstable.

One obvious solution to solving this problem is to introduce more samples of class i into the representation of $\boldsymbol{y}$, yet the problem is how to find the additional samples. Fortunately, one fact in FR is that the face images of different people share certain similarities, and some subjects, say subject i and subject j, can be very similar to each other so that the samples from the class j can be used to represent the testing sample of class i. In other words, one class can borrow samples from the classes similar to it in order to faithfully represent the testing sample. Such a strategy is very similar to the nonlocal technique widely used in image restoration [207-209], where for a given image local patch many similar patches to it (i.e., the so-called nonlocal similar patches) are collected in the image to help the reconstruction of the given patch. By exploiting the nonlocal redundancy, the nonlocal methods achieve state-of-the-art results in the image restoration literature. In FR, for each class we may consider the similar samples from other classes as the "nonlocal samples" and use them to reconstruct the testing sample for a more accurate representation.

However, such a "nonlocal" strategy has some problems to implement under the

scenario of FR. First, how to find the "nonlocal" samples for each class is itself a nontrivial problem. Note that here our goal is face classification but not face reconstruction (though reconstruction is an intermediate stage in the whole classification process), and using the Euclidian/cosine distance to identify the nonlocal samples may not be effective for our goal. Second, by introducing the nonlocal samples to represent the testing sample, all the classes will reduce its representation residual of the testing sample, and thus the discrimination power of representation residual may be reduced, making the classification harder. Third, such a strategy can be computationally expensive because for each class we need to identify the nonlocal samples and calculate the representation. Therefore, we need to find another way to solve the small-sample-size problem.

Interestingly, in SRC [102] this "lack of samples" problem is solved by using the collaborative representation strategy, i.e., coding the testing image $\boldsymbol{y}$ over the samples from all classes $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K]$ as $\boldsymbol{y} \approx \boldsymbol{A}\boldsymbol{\alpha}$. Such a collaborative representation strategy simply takes the face images from all the other classes as the nonlocal samples of one class. Though this representation strategy is very simple, there are two key points that we would like to stress. First, by collaborative representation the searching for the nonlocal samples of each class can be avoided. Second, by collaborative representation all the classes share one common representation of the testing sample, and thus the conventional representation residual based classification procedure, which is used in NN and NS classifiers, cannot be used.

Though we call the representation of $\boldsymbol{y}$ by $\boldsymbol{A}$ "collaborative representation", we have no objection if anyone call it "competitive representation", because each class will contribute competitively to represent $\boldsymbol{y}$. If one class contributes more, this means that other classes will contribute less. In this face representation problem, "collaboration" and "competition" are the two sides of the same coin. Therefore, one intuitive but very effective classification rule is to check which class contributes the most in the collaborative representation of $\boldsymbol{y}$, or equivalently which class has the least reconstruction residual by using the coding coefficients associated with it. This rule is adopted in the

SRC scheme and shows very powerful classification capability. Next, let's make more discussions on this classifier, which can be generally called the collaborative representation based classification (CRC) scheme.

3.3.1 Discussions on collaborative representation based classification

After collaboratively represent $\boldsymbol{y}$ using $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right\}$, SRC classifies $\boldsymbol{y}$ by checking the representation residual class by class using $\text{identity}(\boldsymbol{y}) = \arg\min_i \|\boldsymbol{y} - \boldsymbol{A}_i \hat{\boldsymbol{\alpha}}_i\|_2$, where $\hat{\boldsymbol{\alpha}}_i$ is the sub-coefficient vector associated with class *i*. For the simplicity of analysis, let's remove the l_1-regularization term (i.e., $\|\boldsymbol{\alpha}\|_1$), and the representation becomes the least square problem: $\{\hat{\boldsymbol{\alpha}}_i\} = \arg\min_{\{\boldsymbol{\alpha}_i\}} \left\| \boldsymbol{y} - \sum_i \boldsymbol{A}_i \boldsymbol{\alpha}_i \right\|_2^2$. Refer to Fig. 3.3, the resolved representation $\hat{\boldsymbol{y}} = \sum_i \boldsymbol{A}_i \hat{\boldsymbol{\alpha}}_i$ is the perpendicular projection of $\boldsymbol{y}$ onto the space spanned by $\boldsymbol{A}$. The reconstruction residual by each class is $r_i = \|\boldsymbol{y} - \boldsymbol{A}_i \hat{\boldsymbol{\alpha}}_i\|_2^2$. It can be readily derived that

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

Obviously, when we use r_i to determine the identity of $\boldsymbol{y}$, it is the amount

$$r_i^* = \|\hat{\boldsymbol{y}} - \boldsymbol{A}_i \hat{\boldsymbol{\alpha}}_i\|_2^2 \tag{3-4}$$

that works for classification because $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2$ is a constant for all classes.

From a geometric viewpoint, we can write r_i^* as

$$r_i^* = \frac{\sin^2(\hat{\boldsymbol{y}}, \boldsymbol{\chi}_i) \| \hat{\boldsymbol{y}} \|_2^2}{\sin^2(\boldsymbol{\chi}_i, \bar{\boldsymbol{\chi}}_i)} \tag{3-5}$$

where $\boldsymbol{\chi}_i = \boldsymbol{A}_i \hat{\boldsymbol{\alpha}}_i$ is a vector in the space spanned by $\boldsymbol{A}_i$, and $\bar{\boldsymbol{\chi}}_i = \sum_{j \neq i} \boldsymbol{A}_j \hat{\boldsymbol{\alpha}}_j$ is a vector in the space spanned by all the other classes $\boldsymbol{A}_j$, $j \neq i$. Eq. (3-5) shows that by using CRC, when we judge if $\boldsymbol{y}$ belongs to class *i*, we will not only consider if the angle between $\hat{\boldsymbol{y}}$

and χ_i is small (i.e., if $\sin(\hat{y}, \chi_i)$ is small), we will also consider if the angle between χ_i and $\bar{\chi}_i$ is big (i.e., if $\sin(\chi_i, \bar{\chi}_i)$ is big). Such a "double checking" mechanism makes the CRC effective and robust for classification.

When the number of classes is too big, the number of atoms in dictionary $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K]$ will be big so that the least square solution $\{\hat{\boldsymbol{\alpha}}_i\} = \min_{\{\boldsymbol{\alpha}_i\}} \left\| \boldsymbol{y} - \sum_i \boldsymbol{A}_i \boldsymbol{\alpha}_i \right\|_2^2$ is not unique (i.e., the solution may become unstable). This problem can be solved by regularization. In SRC, the l_1-norm sparsity constraint is imposed on $\boldsymbol{\alpha}$ to regularize the solution. However, the l_1-minimization is time consuming. As we will see in the section of experimental results, by using l_2-norm to regularize the solution of $\boldsymbol{\alpha}$, we can have similar FR results to those by l_1-regularization but with significantly less complexity. This validates that the collaborative representation plays a more important role than the l_1-norm regularization in the problem of FR.

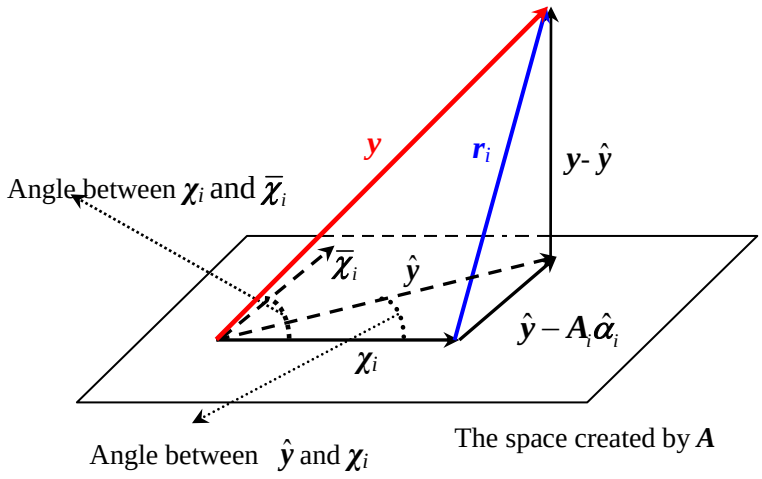


Figure 3.3: Illustration of collaborative representation based classification.

3.3.2 General model of collaborative representation

By coding a given testing image $\boldsymbol{y}$ over the dictionary $\boldsymbol{A}$, we may write it as $\boldsymbol{y}=\boldsymbol{x}+\boldsymbol{e}$, where $\boldsymbol{x}\approx\boldsymbol{A}\boldsymbol{\alpha}$ is the component we want to recover from $\boldsymbol{y}$ for classification and $\boldsymbol{e}$ is the

component (e.g., noise, occlusion and corruption) we want to remove from $\boldsymbol{y}$. A general model of collaborative representation is:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \left\| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha} \right\|_{l_q} + \lambda \left\| \boldsymbol{\alpha} \right\|_{l_p} \right\} \tag{3-6}$$

where λ is the regularization parameter and p, q = 1 or 2. Different settings of p and q lead to different instantiations of the collaborative representation model. For example, in SRC [102] p is set as 1 while q is set as 1 or 2 to handle face recognition with and without occlusion/corruption, respectively.

Different from the sparse representation in image restoration, where the goal is to faithfully reconstruct the signal from the noisy and incomplete observation, in CRC the goal of collaborative representation is twofold. First, we want to recover the desired signal $\boldsymbol{x}$ from observation $\boldsymbol{y}$ with the resolved coding vector $\hat{\boldsymbol{\alpha}}$, i.e., $\boldsymbol{x}=\boldsymbol{A}\hat{\boldsymbol{\alpha}}$. In $\boldsymbol{x}$, the noisy and trivial information are expected to be suppressed. Second, in order for an accurate classification, the coding vector $\hat{\boldsymbol{\alpha}}$ should be sparse enough so that the identity of $\boldsymbol{y}$ can be easily identified. Next let's discuss how to set p and q in Eq. (3-6) to achieve the above goals with a reasonable degree of computational complexity.

Suppose that there is no occlusion/corruption in $\boldsymbol{y}$ (the case that $\boldsymbol{y}$ is occluded/corrupted will be discussed in Section 3.3.4), we may assume that the observed image $\boldsymbol{y}$ contains some additive Gaussian noise. Under such an assumption, it is known that the l_2-norm should be used to characterize the fidelity term in Eq. (3-6) in order for an optimal *maximum a posterior* (MAP) estimation of $\boldsymbol{x}$ [138]. Thus we have q=2.

Let's then discuss the regularization term in Eq. (3-6). Most of the previous works [102, 147, 182] like SRC emphasize the importance of l_1-regularization on $\boldsymbol{\alpha}$, and it is believed that the l_1-regularization on $\boldsymbol{\alpha}$ makes the resolved coding vector $\hat{\boldsymbol{\alpha}}$ sparse. In order to make clearer which norm we should use to regularize $\boldsymbol{\alpha}$, let's conduct some experiments to investigate its distribution.

We use the Extended Yale B and AR [212] databases to perform the experiments. The training samples (1216 samples in Extended Yale B and 700 samples in AR) are used as

the dictionary $\boldsymbol{A}$. The PCA is used to reduce the dimensionality of face images. For each testing face sample $\boldsymbol{y}$, it is coded over $\boldsymbol{A}$, and the coding vector $\boldsymbol{\alpha}$ calculated from all the testing samples are used to draw the histogram of $\boldsymbol{\alpha}$. In the first experiment, we reduce the feature dimensionality of face images to 800 for Extended Yale B and 500 for AR. Then the dictionaries $\boldsymbol{A}$ for the two databases are of size 800×1216 and 500×700, respectively. Since both the two systems are under-determined, we calculate the coding vector with least-square method but with a weak regularization: $\boldsymbol{\alpha} = \left(\boldsymbol{A}^T\boldsymbol{A} + 0.0001\cdot\boldsymbol{I}\right)^{-1}\boldsymbol{A}^T\boldsymbol{y}$. In Figs. 3.4(a) and 3.4(b) we draw the histograms of $\boldsymbol{\alpha}$ for the two databases, as well as the fitted curves of them by using Gaussian and Laplacian functions.

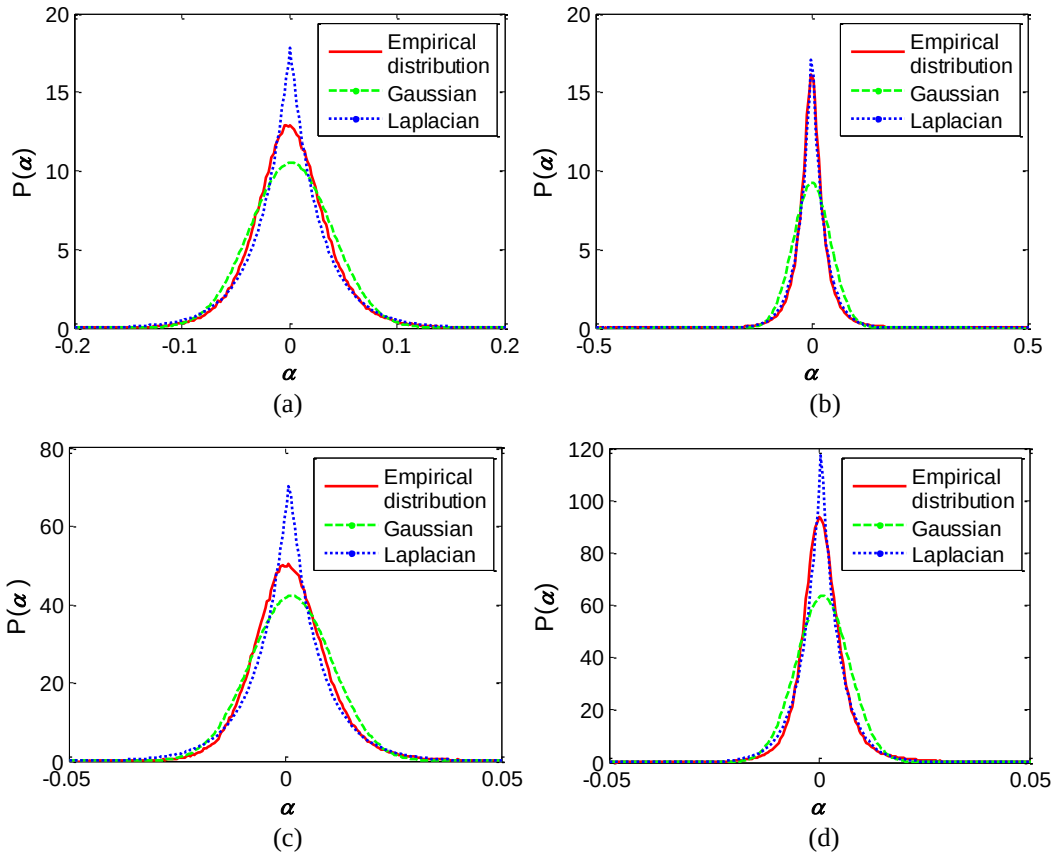


Figure 3.4: The histograms (in red) of the coding coefficients and the fitted curves of them by using Gaussian (in green) and Laplacian (in blue) functions. (a) and (b) show the curves for AR (500-d) and Extended Yale B (800-d) databases, respectively, while (c) and (d) show the curves when the feature dimension is 50.

From Figs. 3.4(a) and 3.4(b), we can see that the distribution of $\boldsymbol{\alpha}$ can be much better fitted as Laplacian than Gaussian. The Kullback-Leibler divergences between the histograms and the fitted curves are 0.0223 by Gaussian and 0.0172 by Laplacian for the AR database, and 0.1071 by Gaussian and 0.0076 by Laplacian for the Extended Yale B database. In other words, via collaborative representation the distribution of $\boldsymbol{\alpha}$ naturally and *passively* tends to be sparse (i.e., Laplacian) even without the l_1-regularization. This is because when the dimension of the face feature $\boldsymbol{y}$ is relatively high (e.g., 500), in general the discrimination power of $\boldsymbol{y}$ is also high so that only a few training samples, mostly from the same class as $\boldsymbol{y}$, will be chosen to code it. This naturally leads to a sparse representation of $\boldsymbol{y}$.

Then we reduce the face feature dimensionality to 50 by PCA, and draw in Figs. 3.4(c) and 3.4(d) the histograms of $\boldsymbol{\alpha}$ on the two databases, as well as the fitted curves of them. It can be found that the Laplacian fitting of the histogram is not that accurate now (the Kullback-Leibler divergences are 0.0264 for the AR database and 0.0152 for the Extended Yale B database), while the Gaussian fitting of the histogram is much improved (the Kullback-Leibler divergences are 0.0231 for the AR database and 0.0820 for the Extended Yale B database). This is because when the dimension of the face feature $\boldsymbol{y}$ is low (e.g., 50), the discrimination capability of $\boldsymbol{y}$ will be much decreased so that quite a few training samples from different classes will be chosen to code $\boldsymbol{y}$. This makes the representation of $\boldsymbol{y}$ much less sparse, and raises the difficulty to correctly identify the identity of $\boldsymbol{y}$.

For a more comprehensive observation of the relationship between the dimensionality of feature $\boldsymbol{y}$ and the sparsity of coding coefficient $\boldsymbol{\alpha}$, in Fig. 3.5 we show the Kullback-Leibler divergences between the coding coefficient histograms and the fitted Gaussian and Laplacian functions under various feature dimensions. Clearly, with the increase of feature dimensionality, the fitting error by Laplacian function decreases, implying that the increase of feature discrimination can naturally force the coding coefficients to be sparsely distributed. In such case, there is no necessary to further

regularize $\boldsymbol{\alpha}$ by using the expensive l_1-norm regularization. However, with the decrease of feature dimensionality, the discrimination power of the feature vector will also decrease and the distribution of $\boldsymbol{\alpha}$ becomes less sparse. In such case, we may need to impose the l_1-regularization on $\boldsymbol{\alpha}$ to *actively* sparsify $\boldsymbol{\alpha}$ to enhance the classification capability. Our experiments in Section 3.4 will also validate the above analyses.

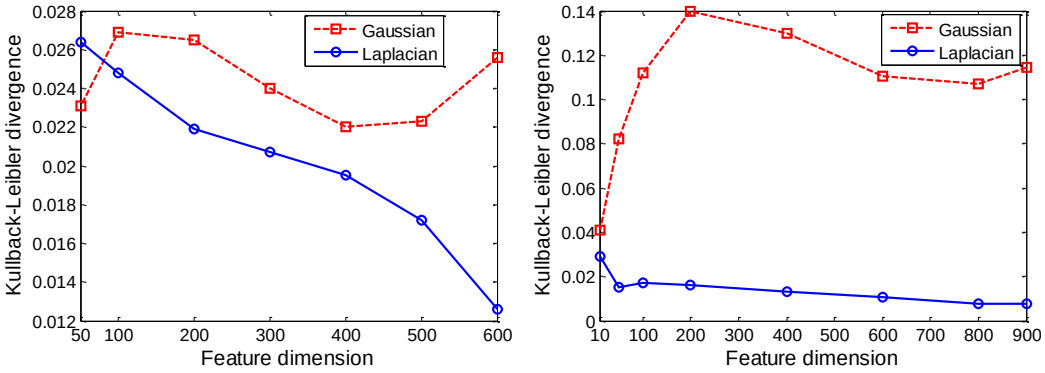


Figure 3.5: The Kullback-Leibler divergences between the coding coefficient histograms and the fitted curves (by Gaussian and Laplacian distributions) under different feature dimensions. (a) AR; and (b) Extended Yale B.

3.3.3 CRC with regularized least square

In practical FR systems, usually the feature dimensionality will not be set too low in order for a good recognition rate. Based on our above discussions, there is no necessary to use l_1-norm to sparsify $\boldsymbol{\alpha}$. Considering that the dictionary $\boldsymbol{A}$ can be under-determined, we use $\|\boldsymbol{\alpha}\|_2$ to regularize the solution of Eq. (3-6), leading to the following regularized least square (RLS) instantiation of collaborative representation:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_2^2 \right\} \tag{3-7}$$

The role of the l_2-norm regularization term $\|\boldsymbol{\alpha}\|_2$ is two-folds. First, it makes the least square solution stable, particularly when $\boldsymbol{A}$ is under-determined; second, it introduces a certain amount of sparsity to the solution $\hat{\boldsymbol{\alpha}}$, yet this sparsity is much weaker than that by l_1-norm.

The solution of RLS based collaborative representation in Eq. (3-7) can be

analytically derived as $\hat{\boldsymbol{\alpha}} = \boldsymbol{P}\boldsymbol{y}$, where $\boldsymbol{P} = \left(\boldsymbol{A}^T\boldsymbol{A} + \lambda \cdot \boldsymbol{I}\right)^{-1}\boldsymbol{A}^T$. Clearly, $\boldsymbol{P}$ is independent of $\boldsymbol{y}$ so that it can be pre-calculated. Once a testing sample $\boldsymbol{y}$ comes, we can simply project $\boldsymbol{y}$ onto $\boldsymbol{P}$ via $\boldsymbol{P}\boldsymbol{y}$. This makes the calculation very fast. The classification by $\hat{\boldsymbol{\alpha}}$ is similar to that in SRC (i.e., $\text{identity}(\boldsymbol{y}) = \arg\min_i \|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i\|_2$). In addition to use the class-specified representation residual $\|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i\|_2$ for classification, where $\hat{\boldsymbol{\alpha}}_i$ is the coding vector associated with class *i*, the l_2-norm "sparsity" $\|\hat{\boldsymbol{\alpha}}_i\|_2$ also brings some discrimination information. We propose to use both of them in the decision making. (Based on our experiments, this improves slightly the classification accuracy over that by using only $\|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i\|_2$.) The proposed CRC algorithm via RLS (CRC-RLS) is summarized in Table 3.1.

Table 3.1: The CRC-RLS Algorithm.

The CRC-RLS Algorithm
1. Normalize the columns of $\boldsymbol{A}$ to have unit l_2-norm.
2. Code $\boldsymbol{y}$ over $\boldsymbol{A}$ by $$\hat{\boldsymbol{\alpha}} = \boldsymbol{P}\boldsymbol{y} \quad (3\text{-}8)$$ where $\boldsymbol{P} = \left(\boldsymbol{A}^T\boldsymbol{A} + \lambda \cdot \boldsymbol{I}\right)^{-1}\boldsymbol{A}^T$.
3. Compute the regularized residuals $$r_i = \|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i\|_2 / \|\hat{\boldsymbol{\alpha}}_i\|_2 \quad (3\text{-}9)$$
4. Output the identity of $\boldsymbol{y}$ as $$\text{identity}(\boldsymbol{y}) = \arg\min_i \{r_i\} \quad (3\text{-}10)$$

3.3.4 Robust CRC (R-CRC) to occlusion/corruption

In Section 3.3.3, we considered the problem of FR without face occlusion/corruption and used l_2-norm to model the coding residual in CRC. However, when there are outliers (e.g., occlusions and corruptions) in the testing face images, using l_1-norm to measure the representation fidelity is more robust than l_2-norm because l_1-norm could tolerate the

outliers. In the robust version of SRC (R-SRC), the l_1-norm is used to measure the coding residual for robustness to occlusions/corruption. Here we could also adopt the l_1-norm coding residual in the CRC scheme for FR with occlusion/corruption, leading to the robust CRC (R-CRC) model:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_1 + \lambda \|\boldsymbol{\alpha}\|_2^2 \right\} \tag{3-11}$$

Let $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\alpha}$. Eq. (3-11) can be re-written as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{e}\|_1 + \lambda \|\boldsymbol{\alpha}\|_2^2 \right\} \quad \text{s.t. } \boldsymbol{y}=\boldsymbol{A}\boldsymbol{\alpha}+\boldsymbol{e} \tag{3-12}$$

Eq. (3-12) is a constrained convex optimization problem which can be efficiently solved by the Augmented Lagrange Multiplier (ALM) method [210-211]. The corresponding augmented Lagrange function is given by

$$L_\mu(\boldsymbol{e}, \boldsymbol{\alpha}, \boldsymbol{z}) = \|\boldsymbol{e}\|_1 + \lambda \|\boldsymbol{\alpha}\|_2^2 + \langle \boldsymbol{z}, \boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha} - \boldsymbol{e} \rangle + \frac{\mu}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha} - \boldsymbol{e}\|_2^2 \tag{3-13}$$

where μ>0 is a constant that determines the penalty for large representation error, and $\boldsymbol{z}$ is a vector of Lagrange multipliers. The ALM algorithm iteratively estimates the Lagrange multipliers and the optimal solution by iteratively minimizing the augmented Lagrangian function

$$(\boldsymbol{e}_{k+1}, \boldsymbol{\alpha}_{k+1}) = \arg\min_{\boldsymbol{e},\boldsymbol{\alpha}} L_{\mu_k}(\boldsymbol{e}, \boldsymbol{\alpha}, \boldsymbol{z}_k) \tag{3-14}$$

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \mu_k (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha} - \boldsymbol{e}) \tag{3-15}$$

The above iteration could converge to the optimal solution of Eq. (3-12) when $\{\mu_k\}$ is a monotonically increasing positive sequence [210].

The minimization in the first stage (i.e., Eq. (3-14)) of the ALM iteration could be implemented by alternatively and iteratively updating the two unknowns $\boldsymbol{e}$ and $\boldsymbol{\alpha}$ as follows:

$$\begin{cases} \boldsymbol{\alpha}_{k+1} = \arg\min_{\boldsymbol{\alpha}} L_{\mu_k}(\boldsymbol{\alpha}, \boldsymbol{e}_k, \boldsymbol{z}_k) \\ \boldsymbol{e}_{k+1} = \arg\min_{\boldsymbol{e}} L_{\mu_k}(\boldsymbol{\alpha}_{k+1}, \boldsymbol{e}, \boldsymbol{z}_k) \end{cases} \tag{3-16}$$

for which we could have a closed-form solution:

$$\begin{cases} \boldsymbol{\alpha}_{k+1} = \left(\boldsymbol{A}^T\boldsymbol{A} + 2\lambda/\mu_k\boldsymbol{I}\right)^{-1}\boldsymbol{A}^T\left(\boldsymbol{y} - \boldsymbol{e}_k + \boldsymbol{z}_k/\mu_k\right) \\ \boldsymbol{e}_{k+1} = S_{1/\mu_k}\left[\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}_{k+1} + \boldsymbol{z}_k/\mu_k\right] \end{cases} \tag{3-17}$$

where the function S_α , $\alpha \geq 0$, is the soft-thresholding or shrinkage operator defined component-wise as

$$\left[S_\alpha(\boldsymbol{x})\right]_i = \text{sign}(x_i)\cdot\max\left\{|x_i| - \alpha, 0\right\} \tag{3-18}$$

Clearly, $\boldsymbol{P}_k = \left(\boldsymbol{A}^T\boldsymbol{A} + 2\lambda/\mu_k\boldsymbol{I}\right)^{-1}\boldsymbol{A}^T$ is independent of $\boldsymbol{y}$ for the given μ_k and thus $\{\boldsymbol{P}_k\}$ can be pre-calculated as a set of projection matrices. Once a testing sample $\boldsymbol{y}$ comes, in the first stage of ALM we can simply project $\boldsymbol{y}$ onto $\boldsymbol{P}_k$ via $\boldsymbol{P}_k\boldsymbol{y}$. This makes the calculation very fast. After solving the representation coefficients $\boldsymbol{\alpha}$ and residual $\boldsymbol{e}$, similar classification strategy to CRC-RLS can be adopted by R-CRC. The entire algorithm of R-CRC is summarized in Table 3.2.

Table 3.2: The R-CRC Algorithm.

The R-CRC Algorithm
1. Normalize the columns of $\boldsymbol{A}$ to have unit l_2-norm.
2. Code $\boldsymbol{y}$ over $\boldsymbol{A}$ by
INPUT: $\boldsymbol{\alpha}_0$, $\boldsymbol{e}_0$ and $\tau > 0$.
WHILE not converged **Do**
$\boldsymbol{\alpha}_{k+1} = \left(\boldsymbol{A}^T\boldsymbol{A} + 2\lambda/\mu_k\boldsymbol{I}\right)^{-1}\boldsymbol{A}^T\left(\boldsymbol{y} - \boldsymbol{e}_k + \boldsymbol{z}_k/\mu_k\right)$
$\boldsymbol{e}_{k+1} = S_{1/\mu_k}\left[\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}_{k+1} + \boldsymbol{z}_k/\mu_k\right]$
$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \mu_k\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}_{k+1} - \boldsymbol{e}_{k+1}\right)$
End WHILE
OUTPUT: $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{e}}$.
3. Compute the regularized residuals
$r_i = \left\|\boldsymbol{y} - \boldsymbol{A}_i\hat{\boldsymbol{\alpha}}_i - \hat{\boldsymbol{e}}\right\|_2 / \left\|\hat{\boldsymbol{\alpha}}_i\right\|_2$
4. Output the identity of $\boldsymbol{y}$ as
$\text{identity}(\boldsymbol{y}) = \arg\min_i \{r_i\}$

3.4 Experimental Results

In the experiments from Sections 3.4.1 to 3.4.4, considering the accuracy and computational efficiency we chose l_1_ls [204] to solve the l_1-regularized SRC scheme. In the experiments, we denote by S-SRC the standard SRC (i.e., the coding residual is measured by l_2-norm) and by R-SRC the robust version of SRC (i.e., the coding residual is measured by l_1-norm). All the experiments were implemented using MATLAB on a 3.16 GHz machine with 3.25GB RAM. In our paper, the parameter λ of CRC-RLS and RNS_L_p (p=1 or 2) in gender classification is set as 0.08. Considering that in FR when more classes (and thus more samples) are used for collaborative representation, the least square solution will be more unstable and thus higher regularization is requires, we set λ as $0.001 \cdot n/700$ for CRC_RLS in all FR experiments, where n is the number of training samples. If there is no specific instruction, for R-CRC we set λ as 1 in FR with occlusion. Three benchmark face databases, the Extended Yale B [99, 206], AR [212] and Multi-PIE [213], are used in the evaluation of CRC and its competing methods, including SRC, SVM, LRC [101], and NN. (Note that LRC is an NS based method.)

The experiments are arranged as follows. In Section 3.4.1, we use examples to discuss the role of l_1-norm and l_2-norm regularization; in Section 3.4.2, we use gender classification as an example to illustrate that collaborative representation is not necessary when there are enough training samples of each class; then FR without and with occlusion/corruption are conducted in Section 3.4.3 and Section 3.4.4, respectively; finally the running time of SRC and CRC is evaluated in Section 3.4.5.

3.4.1 L_1-regularization vs. L_2-regularization

In this section, we study the role of sparsity constraint in FR. Here we use the Extended Yale B [99, 206] and AR [212] for experiments (the experimental setting will be described in Section 3.4.3). The Eigenfaces with dimensionality 300 are used as the input facial

features. The dictionary is formed by all the training samples.

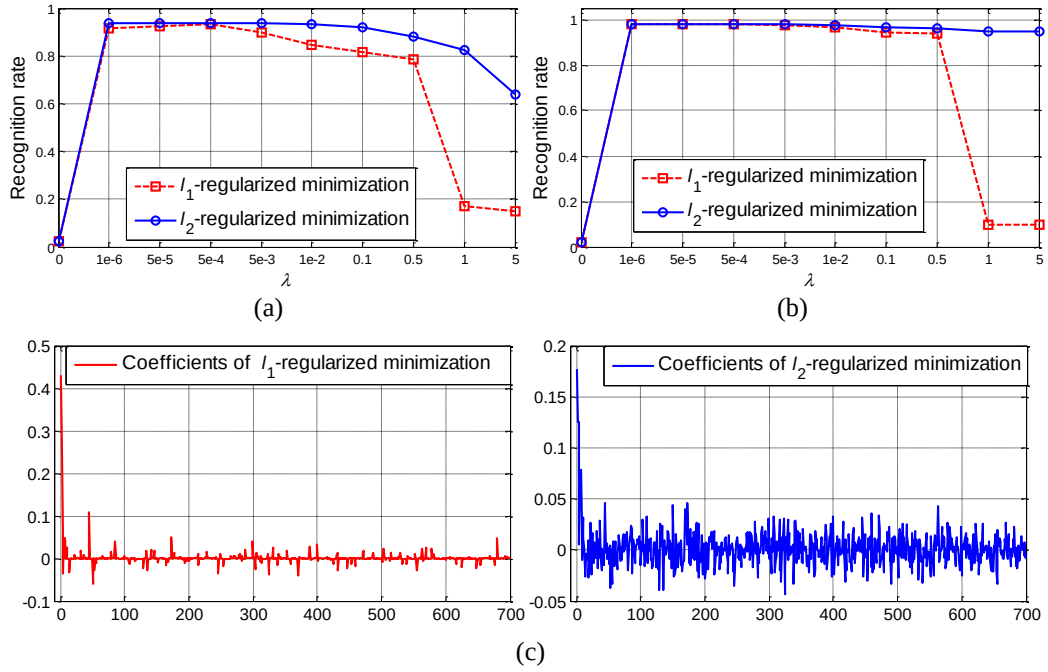


Figure 3.6: The recognition rates of S-SRC (l_1-regularized minimization) and CRC-RLS (l_2-regularized minimization) versus the different values of λ on the (a) AR and (b) Extended Yale B databases. The coding coefficients of one testing sample are plotted in (c).

We test the performance of S-SRC (l_1-regularized minimization) and CRC-RLS (l_2-regularized minimization) by increasing the regularization parameter λ in Eq. (3-6) with q=2, p=1 (i.e., S-SRC) and q=2, p=2 (CRC_RLS). The results on the AR and Extended Yale B databases are shown in Fig. 3.6(a) and Fig. 3.6(b), respectively. We can see that when λ=0, both S-SRC and CRC-RLS will fail. When λ is assigned a small positive value, e.g., from 0.000001 to 0.1, good results can be achieved by S-SRC and CRC-RLS. When λ is too big (e.g., >0.1) the recognition rates of both methods fall down. From Fig. 3.6 we can find that with the increase of λ (>0.000001), no much benefit on recognition rate can be gained. In addition, the l_2-regularized minimization (i.e., CRC-RLS) could get similar or slightly higher recognition rates than the l_1-regularized minimization (i.e., S-SRC) in a broad range of λ. This validates our discussion in Section 3.3.2 that the l_1-regularization on

$\boldsymbol{\alpha}$ is not necessary when the discrimination of face feature is high enough, and the l_2-regularization on $\boldsymbol{\alpha}$ is as effective as l_1-regularization in classification. However, when the dimension of facial features is very low, the representation will become very under-determined, and the FR results by l_1-norm and l_2-norm regularizations could be substantially different, as demonstrated in [214] and discussed in Section 3.3.2 of this Chapter. In such case, l_1-regularization is helpful to get discriminative coefficients for accurate FR.

Fig. 3.6(c) plots one testing sample's coding coefficients by S-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 S-SRC; however, it could achieve no worse FR results. Again, l_1-sparsity is useful but is not crucial for FR. What is really crucial is the collaborative representation mechanism in CRC-RLS and S-SRC.

3.4.2 Gender classification

In Section 3.2, we indicated that when the number of samples in each class is big enough, there is no need to code the testing sample over the samples from all classes because the subset of each class can form a nearly over-complete dictionary already. To validate this claim, we conduct experiment on a two-class separation problem: gender classification. We chose a non-occluded subset (14 images per subject) of AR [212], which consists 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. PCA is used to reduce the dimension of each image to 300. Since there are enough training samples in each class, as we discussed in Section 3.2, the RNS_L_p (refer to Eq. (3-3) and the related explanations) methods should do a good job for the classification task.

We compare RNS_L_1 and RNS_L_2 with the CRC-RLS, S-SRC, SVM, LRC, and NN methods. The results are listed in Table 3.3. One can see that RNS_L_1 and RNS_L_2 get the same best results, validating that coding on each class' dictionary is more powerful than

coding on the whole dictionary when the training samples are enough, no matter l_1- or l_2-regularizaion is used. CRC-RLS gets the second best result, about 1.4% higher than S-SRC. This experiment also shows that the regularization is very helpful to improve the classification accuracy.

Table 3.3: The results of different methods on gender classification using the AR database.

RNS_L_1	RNS_L_2	CRC-RLS	S-SRC	SVM	LRC	NN
94.9%	**94.9%**	93.7%	92.3%	92.4%	27.3%	90.7%

3.4.3 Face recognition without occlusion/corruption

We then test the proposed CRC-RLS method for FR without occlusion/corruption on the benchmark Extended Yale B, AR and MPIE face databases. The Eigenface is used as face feature in these experiments.

1) Extended Yale B Database: The Extended Yale B [99, 206] 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 3.4 shows the recognition rates versus feature dimension by NN, LRC, SVM, S-SRC and CRC-RLS. Here we also report the performance of RNS_L_2 due to the high performance of RNS_L_p in gender classification. It can be seen that the best two methods, CRC-RLS and S-SRC, achieve very similar recognition rates. When the feature dimensionality is relatively high (e.g., 150 and 300), the difference of their recognition rate is less than 0.5%. When the feature dimensionality is set very low (e.g., 50), S-SRC will show some advantage over CRC-RLS in terms of recognition rate. This is exactly in accordance with our analysis in Section 3.3.2. We also see that RNS_L_2 has lower recognition accuracy than CRC_RLS

when the dimension is not too low (e.g., >50). Since there are enough (about 32 per class) training samples in the Extended Yale B database, all the methods show no bad recognition rates in this experiment.

2) AR database: As in [102], a subset (with only illumination and expression changes) that contains 50 male subjects and 50 female subjects was chosen from the AR dataset [212] in our experiments. For each subject, the seven images from Session 1 were used for training, with other seven images from Session 2 for testing. The size of image was cropped to 60×43. The comparison of competing methods is given in Table 3.5. We can see that CRC-RLS achieves the best result when the dimensionality is 120 or 300, while it is slightly worse than S-SRC when the dimensionality is very low (e.g., 54). This is again in accordance with our analysis in Section 3.3.2. The recognition rates of CRC-RLS and S-SRC are both at least 10% higher than other methods, including RNS_L_2, which has similar performance to LRC. This shows that collaborative representation do improve much face classification accuracy.

Table 3.4: The face recognition results of different methods on the Extended Yale B database.

Dim	50	150	300
NN	78.5%	90.0%	91.6%
LRC	93.1%	95.1%	95.9%
RNS_L_2	**94.6%**	95.8%	96.3%
SVM	93.4%	96.4%	97.0%
S-SRC	93.8%	**96.8%**	**97.9%**
CRC-RLS	92.5%	96.3%	**97.9%**

Table 3.5: 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%
RNS_L_2	70.2%	74.8%	76.1%
SVM	69.4%	74.5%	75.4%
S-SRC	**83.3%**	89.5%	93.3%
CRC-RLS	80.5%	**90.0%**	**93.7%**

Table 3.6: The face recognition results of different methods on the MPIE database.

	NN	LRC	SVM	S-SRC	CRC-RLS
Session 2	86.4%	87.1%	85.2%	93.9%	**94.1%**
Session 3	78.8%	81.9%	78.1%	**90.0%**	89.3%
Session 4	82.3%	84.3%	82.1%	**94.0%**	93.3%

3) Multi PIE database: The CMU Multi-PIE database [213] contains images of 337 subjects captured in four sessions with simultaneous variations in pose, expression, and illumination. Among these 337 subjects, all the 249 subjects in Session 1 were used. For the training set, we used the 14 frontal images with 14 illuminations[2] with 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 3.6 lists the recognition rates in three tests by the competing methods. The results validate that CRC-RLS and S-SRC are the two best methods in accuracy, and they have at least 6% improvement over the other three methods.

3.4.4 Face recognition with occlusion/corruption

One of the most interesting features of representation (or coding) based FR methods is their ability to deal with occlusion and corruptions. In R-SRC [102], the robustness to face occlusion/corruption is achieved by adding an occlusion dictionary (an identity matrix) for sparse coding, or equivalently, using l_1-norm to measure the coding residual. In Section 3.3.4, we have correspondingly presented the robust version of CRC, i.e., R-CRC, for FR with occlusion/corruption. In this section we evaluate the performance of R-CRC to handle different kinds of occlusions, including random pixel corruption, random block occlusion and real disguise. The results of CRC-RLS are also presented for comparison.

1) FR with block occlusion: To be identical to the experimental settings in [102], we

[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}.

used Subsets 1 and 2 (717 images, normal-to-moderate lighting conditions) of the Extended Yale B database for training, and used Subset 3 (453 images, more extreme lighting conditions) for testing. As in [102], we simulate various levels of contiguous occlusion, from 0% to 50%, by replacing a randomly located square block of each testing image with an unrelated image. The block occlusion of a certain size is located on the random position which is unknown to the FR algorithms. The images were resized to 96×84. Here λ of R-CRC is set as 0.1. The results by S-SRC, R-SRC, CRC-RLS and R-CRC are shown in Table 3.7. We can see that R-CRC outperforms R-SRC in most cases (with 17% improvement in 50% occlusion) except for the case of 30% block occlusion. In addition, CRC-RLS could achieve much better performance than S-SRC. This is mainly because the testing sample with block occlusion cannot be well represented by the non-occluded samples with sparse coefficients. In the following experiments, we only report the results of R-SRC in FR with corruption or disguise.

Table 3.7: The recognition rates of R-CRC, CRC-RLS, R-SRC and S-SRC under different levels of block occlusion.

Occlusion	0%	10%	20%	30%	40%	50%
S-SRC	**100%**	99.6%	93.4%	77.5%	60.9%	45.9%
R-SRC	**100%**	**100%**	99.8%	**98.5%**	90.3%	65.3%
CRC-RLS	**100%**	**100%**	95.8%	85.7%	72.8%	59.2%
R-CRC	**100%**	**100%**	**100%**	97.1%	**92.3%**	**82.3%**

Table 3.8: The recognition rates (%) of R-SRC, CRC-RLS and R-CRC under different levels of pixel corruption.

Corruption	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
R-SRC	**100**	**100**	**100**	**100**	**100**	**100**	99.3	**90.7**	37.5	7.1
CRC-RLS	**100**	**100**	**100**	99.8	98.9	96.4	79.9	45.7	13.2	4.2
R-CRC	**100**	**100**	**100**	**100**	**100**	**100**	**100**	90.5	**51.0**	**15.9**

2) FR with pixel corruption: In this part, we test the robustness of R-SRC and R-CRC to pixel corruption. We used the same experimental settings as in [102], i.e., Subsets 1 and 2 of Extended Yale B for training and Subset 3 for testing. The images were resized to 96×84 pixels. For each testing image, we replaced a certain percentage of its pixels by

uniformly distributed random values within [0, 255]. The corrupted pixels were randomly chosen for each testing image and the locations are unknown to the algorithm. Table 3.8 lists the recognition rates of R-SRC, CRC-RLS and R-CRC. It can be seen that R-CRC achieves equal or better performance (about 13% improvement over R-SRC in 80% corruption) in almost all cases. Interestingly, CRC-RLS can also perform well up to 50% pixel corruption.

3) FR with real face disguise: As in [102], a subset from the AR database consists of 1,200 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. 3.7) were used for testing. The images were resized to 83×60. The results of competing methods are shown in Table 3.9.

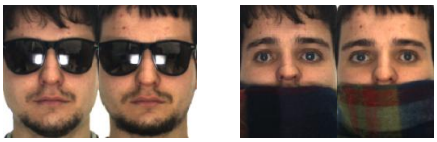


Figure 3.7: The testing samples with sunglasses and scarves in the AR database.

Table 3.9: The results of different methods on face recognition with real disguise (AR database).

	Sunglass	Scarf
R-SRC	**87.0%**	59.5%
CRC-RLS	68.5%	**90.5%**
R-CRC	**87.0%**	**86.0%**
Partitioned	Sunglass	Scarf
R-SRC	**97.5%**	93.5%
CRC-RLS	91.5%	**95.0%**
R-CRC	92.0%	**94.5%**

Although CRC-RLS is not designed for robust FR, interestingly it achieves the best result of FR with scarf disguise, outperforming SRC by a margin of 31% and R-CRC by

4.5%. (The phemonemon that CRC-RLS is better than R-CRC in scarf disguise may result from the special experimental setting, which will be validated by the following FR experiments.) By using l_1-norm to measure the representation fidelity, R-CRC has the same recognition rate as R-SRC in sunglasses disguise, but achieves 26.5% improvement in scarf disguise. As in [102], we also partition the face image into 8 sub-regions for FR. With partition, CRC-RLS and R-CRC can still achieve slightly better performance than R-SRC in scarf disguise, but perform a little worse in sunglass disguise. The reason can be that for each partitioned face portion its discrimination power is limited so that the l_1-regularization is useful to improve the sparsity of coding vector and consequently the classification accuracy. Nevertheless, the recognition rates of CRC-RLS and R-CRC are very competitive with R-SRC.

In the above experiment of FR with scarf, the CRC-RLS model with l_2-norm characterization of coding residual achieves higher recognition rates than the models with l_1-norm characterization of coding residual (i.e., R-SRC and R-CRC), while the reverse is true in the case of FR with sunglasses. To have a more comprehensive observation of these methods' robustness to disguise, we perform another more challenging experiment. A subset from the AR database consists of 1,900 images from 100 subjects, 50 male and 50 female, is used. 700 images (7 samples per subject) of non-occluded frontal views from session 1 were used for training, while all the images with sunglasses (or scarf) from the two sessions were used for testing (6 samples per subject per disguise). The images were resized to 83×60. The FR results are shown in Table 3.10. In this experiment, R-CRC is slightly worse than R-SRC in sunglass case with 4% gap, but significantly better than R-SRC in the scarf case with 32.4% improvement. Compared to R-SRC, CRC-RLS has 31% higher recognition rate in scarf case, and 13% lower rate in sunglass case. It can also be seen that R-CRC achieves better performance than CRC-RLS in these two cases, which validates that the l_1-norm regularization of representation residual is more robust than the l_2-norm one.

Table 3.10: The results on another face recognition experiment with real disguise (AR database).

	Sunglass	Scarf
R-SRC	**69.8%**	40.8%
CRC-RLS	57.2%	**71.8%**
R-CRC	65.8%	**73.2%**

From the results in Table 3.9 and Table 3.10, we can have the following findings. Since the eyes are probably the most discriminative part in human face, the sunglass disguise will reduce a lot the discrimination capability of face image, and hence the l_1-regularized R-SRC method will show certain advantage in dealing with sunglass disguise because the l_1-regularization could actively increase much the sparsity of coding coefficients. (Please refer to Section 3.3.2 for more discussions on the relationship between feature discrimination and coefficient sparsity.) In the case of scarf disguise, though the occlusion area is big, the discrimination of face image is actually not much decreased. Therefore, the l_2-regularized CRC-RLS and R-CRC methods can perform well. On the contrary, the l_1-regularization in R-SRC will prevent the use of enough samples to represent the occluded face image so that its recognition rate is much lower than CRC-RLS and R-CRC.

3.4.5 Running time

We compare the running time of CRC and SRC under two situations. For FR without occlusion/corruption, it is good to use l_2-norm to measure the coding residual, and hence we compare the running time of S-SRC and CRC-RLS; for FR with occlusion/corruption, we compare the running time of R-SRC and R-CRC, where l_1-norm is used to measure the coding residual for robustness to outlier pixels.

a) Face recognition without occlusion: The running time of CRC-RLS and S-SRC with various fast l_1-minimization methods, including l_1_ls [204], ALM [126, 202], FISTA [201] and Homotopy[203], are compared here. We fix the dimensionality of Eigenfaces as

300. The recognition rates and speed of S-SRC and CRC-RLS are listed Table 3.11 (Extended Yale B), Table 3.12 (AR) and Table 3.13 (Multi-PIE), respectively. Note that the results in Table 3.13 are the averaged values of Sessions 2, 3 and 4.

Table 3.11: Recognition rate and speed on the Extended Yale B database.

	Recognition rate	Time (s)
S-SRC(l_1_ls)	**97.9%**	5.3988
S-SRC(ALM)	**97.9%**	0.1280
S-SRC(FISTA)	91.4%	0.1567
S-SRC(Homotopy)	94.5%	0.0279
CRC-RLS	**97.9%**	**0.0033**
Speed-up	**8.5~1636 times**	

Table 3.12: Recognition rate and speed on the AR database.

	Recognition rate	Time (s)
S-SRC(l_1_ls)	93.3%	1.7878
S-SRC(ALM)	93.3%	0.0578
S-SRC(FISTA)	68.2%	0.0457
S-SRC(Homotopy)	82.1%	0.0305
CRC-RLS	**93.7%**	**0.0024**
Speed-up	**12.6~744.9 times**	

Table 3.13: Recognition rate and speed on the MPIE database.

	Recognition rate	Time (s)
S-SRC(l_1_ls)	**92.6%**	21.290
S-SRC(ALM)	92.0%	1.7600
S-SRC(FISTA)	79.6%	1.6360
S-SRC(Homotopy)	90.2%	0.5277
CRC-RLS	92.2%	**0.0133**
Speed-up	**39.7~1600.7 times**	

On the Extended Yale B database, CRC-RLS, S-SRC (l_1_ls) and S-SRC (ALM) achieve the best recognition rate (97.9%), but the speed of CRC-RLS is 1636 and 38.8 times faster than them. On the AR database, CRC-RLS has the best recognition rate and speed. S-SRC (l_1_ls) has the second best recognition rate but with the slowest speed. S-SRC (FISTA) and S-SRC (Homotopy) are much faster than S-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 S-SRC (l_1_*ls*)) but it is significantly (more than 1600 times) faster than S-SRC (l_1_*ls*). In this large-scale database, CRC-RLS is about 40 times faster than S-SRC with the fastest implementation (i.e., Homotopy), while achieving more than 2% improvement in recognition rate. We can see that the speed-up of CRC-RLS is more and more obvious as the scale (i.e., the number of classes or training samples) of face database increases, implying that it is more advantageous in practical large-scale FR applications.

b) Face recognition with occlusion: We compare the time complexity of R-CRC with the latest fast l_1-minimization methods on the Multi-PIE corruption experiment [213]. As in [126] and [215], a subset of 249 subjects from Session 1 is used in this experiment. For each subject with frontal view, there are 20 images with different illuminations, among which the illuminations {0, 1, 7, 13, 14, 16, 18} are chosen as training images with the remaining 13 images as testing data. The images are manually aligned and cropped to 40×30. For each testing image, we replaced a certain percentage of its pixels by uniformly distributed random values within [0, 255]. The corrupted pixels were randomly chosen for each testing image and the locations are unknown to the algorithm. The recognition rates and running time of other competing methods are directly copied from [126, 215]. In order to make a fair comparison of running time, we used a machine similar to that used in [126, 215] to implement R-CRC[4].

Table 3.14 shows the FR rates of R-CRC and R-SRC implemented by various l_1-minimization solvers. One can see that R-CRC has the highest recognition rate in 40% and 50% corruption. In other cases, R-CRC is better than SpaRSA [200] and FISTA [201], and slightly worse than l_1_*ls* [204], Homotopy [203] and ALM [126]. The running time of different methods under various corruption levels is listed in Table 3.15. Apart from the

[4] Our MATLAB implementations are on a PC with dual quad-core 2.4G GHz Xeon processors and 16GB RAM, similar to that used in [126] and [215], in which the machine is with dual quad-core 2.66GHz Xeon processors and 8GB of memory.

case of 0% corruption, the proposed R-CRC has the lowest running time. It can also be seen that the running time of R-CRC is almost the same for all corruption levels. The speed-ups of R-CRC over R-SRC with various l_1-minimization algorithms are from 8.79 to 19.94 in average, showing that R-CRC has much lower time complexity.

Table 3.14: Average recognition rate between 50% and 70% random pixel corruptions on the MPIE database.

Corruption	R-CRC	l_1_*ls*	Homotopy	SpaRSA	FISTA	ALM
40%	**100%**	97.8%	99.9%	98.8%	99.0%	99.9%
50%	**100%**	99.5%	99.8%	97.6%	96.2%	99.5%
60%	94.6%	96.65	**98.7%**	90.5%	86.8%	96.2%
70%	68.4%	76.3%	**84.6%**	63.3%	58.7%	78.8%

Table 3.15: The running time (second) of different methods versus various corruption rate.

Corruption	0%	20%	40%	60%	80%	Average	Speed-up
l_1_*ls*	19.8	18.44	17.47	16.99	14.37	17.35	**18.94**
Homotopy	**0.33**	2.01	4.99	12.26	20.68	8.05	**8.79**
SpaRSA	6.64	10.86	16.45	22.66	23.23	15.97	**17.43**
FISTA	8.78	8.77	8.77	8.80	8.66	8.76	**9.56**
ALM	18.91	18.85	18.91	12.21	11.21	16.02	**17.49**
R-CRC	0.916	**0.914**	**0.918**	**0.916**	**0.915**	**0.916**	------

3.5 Summary

We discussed the role of l_1-norm regularization in the sparse representation based classification (SRC) scheme for face recognition (FR), and we indicated that the collaborative representation nature of SRC plays a more important role than the l_1-regularization of coding vector in face representation and recognition. We then proposed a more general model, namely collaborative representation based classification (CRC), for FR. Two important instantiations of CRC, i.e., CRC via regularized least square (CRC-RLS) and robust CRC (R-CRC), were proposed for FR without and with occlusion/corruption, respectively. Compared with the l_1-norm regularization, the l_2-norm

regularization in CRC has very competitive or even better FR accuracy but with much lower complexity, as demonstrated in our extensive experimental results.

SRC is also an instantiation of CRC by using l_1-norm to regularize the coding vector $\boldsymbol{\alpha}$. The sparsity of $\boldsymbol{\alpha}$ is related to the discrimination and dimension of face feature $\boldsymbol{y}$. If the dimension is high, often the discrimination of $\boldsymbol{y}$ is high and $\boldsymbol{\alpha}$ will be naturally and *passively* sparse even without sparse regularization. In this case, l_1-regularization on $\boldsymbol{\alpha}$ will not show advantage. If the dimension of $\boldsymbol{y}$ is very low, often the discrimination of $\boldsymbol{y}$ is low, and thus it is helpful to *actively* sparsify $\boldsymbol{\alpha}$ by imposing l_1-regularization on it. In this case, using l_1-norm to regularize $\boldsymbol{\alpha}$ will show visible advantage.

Chapter 4. Gabor Feature based Robust Representation and Classification

4.1 Introduction

The high-dimensional facial images usually lie in a lower dimensional subspaces or sub-manifolds. This fact boosts the development of subspace learning and manifold learning based FR methods, such as Eigenface and Fisherface [57-58, 216-217], nonlinear dimension reduction [67-68] as well as its linear approximations [218-220]. In the pioneer work of sparse representation based classification (SRC) [102], the training face images are used to code an input testing image as a sparse linear combination of them via l_1-norm minimization. To make the l_1-norm sparse coding computationally feasible, in general the dimensionality of the training and testing face images should be reduced, or a set of features could be extracted from the original image for SRC. In the case of FR without occlusion, Wright *et al.* [102] tested different types of features, including Eigenface [57], Randomface [102] and Fisherface [58], and they claimed that SRC is insensitive to feature types when the feature dimension is large enough. In the case of FR with occlusion/corruption, an occlusion dictionary was introduced in SRC to code the occluded/corrupted components [102]. Consequently, the classification can be performed based on the reconstruction residuals using the coding coefficients over the training face images. Such a scheme has shown to be effective in overcoming the problem of face occlusion, which triggers the research of sparsity based FR [103, 180, 187-189, 195] and dictionary learning for sparse representation [129, 157, 184-185, 196].

Although the SRC based FR scheme proposed in [102] is very creative and effective, there are two issues to be further addressed. First, the features of Eigenface, Randomface and Fisherface tested in [102] are all holistic features. Since in practice the number of

training samples is often limited, such holistic features cannot effectively handle the variations of illumination, expression, pose and local deformations. The claim made in [102] that feature extraction is not so important to SRC actually holds only for holistic features. Second, the occlusion matrix proposed in [102] is an orthogonal matrix, such as the identify matrix, Fourier bases or Haar wavelet bases, etc. However, the number of atoms required in the orthogonal occlusion matrix is very high. For example, if the dimensionality of features used in SRC is 3000, then a 3000×3000 occlusion matrix is needed. Such a big occlusion matrix makes the sparse coding process very computationally expensive, and even prohibitive. These two issues are not fully solved by the sparsity based FR improvers [103, 147, 157, 180, 184-185, 187-189, 195], either. For instance, only holistic features are considered in [103, 157, 180, 184-185, 187-189, 195], FR with occlusion is ignored in [147, 180, 188], and no occlusion dictionary is considered in [157, 184-185].

In the light of the collaborative representation based classification presented in Chapter 3, we propose a Gabor-feature based robust representation and classification (GRRC) scheme for FR, which will not only be robust to face occlusion but also have much higher computational efficiency than the previous methods such as SRC. In the proposed GRRC, the use of Gabor kernels will not only improve much the FR accuracy, it will also allow us to learn a compact occlusion dictionary to deal with face occlusions. Compared with the occlusion dictionary used in SRC, the number of atoms is significantly reduced (often with a ratio of 40:1 ~ 50:1 in our experiments) in the Gabor occlusion dictionary (GOD) used in GRRC. Particularly, it is found that the coding coefficients over the compact GOD can be regularized by l_2-norm. This significantly reduces the computational cost in coding occluded face images. Our experiments on benchmark face databases clearly validate the performance of the proposed GRRC method.

Table 4.1 summarizes the abbreviations used throughout the Chapter.

Table 4.1: Abbreviation used in this Chapter.

Abbreviation	Meaning
GOD	Gabor Occlusion Dictionary
GRR	Gabor-feature based Robust Representation
GRRC	Gabor-feature based Robust Representation based Classification
GRRC_L$_p$	GRRC with l_p-norm regularization
SRC	Sparse Representation-based Classification
CRC	Collaborative Representation based Classification

4.2 Gabor Features

The Gabor filter was first introduced by David Gabor in 1946 [221], and was later shown as models of simple cell receptive fields [222]. The Gabor filters, which could effectively extract the image local directional features at multiple scales, have been successfully and prevalently used in FR [89, 223-224], leading to state-of-the-art results. The local Gabor features are less sensitive to variations of illumination, expression and pose than the holistic features such as Eigenface and Randomface [102].

The Gabor filters (kernels) with orientation μ and scale v are defined as [89]:

$$\psi_{\mu,v}(z)=\frac{\|k_{\mu,v}\|^2}{\sigma^2}e^{-\|k_{\mu,v}\|^2\|z\|^2/2\sigma^2}\left(e^{ik_{\mu,v}z}-e^{-\sigma^2/2}\right) \tag{4-1}$$

where z=(x,y) denotes the pixel, ||.|| denotes the norm operator, and the wave vector $k_{\mu,v}$ is defined as $k_{\mu,v}=k_v e^{i\phi_\mu}$ with $k_v=k_{max}/f^v$ and $\phi_\mu=\pi\mu/8$. k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain. In addition, σ determines the ratio of the Gaussian window width to wavelength.

Convolving an image *Img* with a Gabor kernel $\psi_{\mu,v}$ outputs $G_{\mu,v}(z)=Img(z)*\psi_{\mu,v}(z)$, where "*" denotes the convolution operator. The complex Gabor filtering coefficient $G_{\mu,v}(z)$ can be rewritten as

$$G_{\mu,v}(z)=M_{\mu,v}(z)\cdot\exp\left(i\theta_{\mu,v}(z)\right)$$

with $M_{\mu,v}$ being the magnitude and $\theta_{\mu,v}$ being the phase. It is known that magnitude

information contains the variation of local energy in the image. In [89], the augmented Gabor feature vector $\boldsymbol{\chi}$ is defined via uniform down-sampling, normalization and concatenation of the Gabor filtering coefficients:

$$\boldsymbol{\chi} = \left(\boldsymbol{a}_{0,0}^{(\rho)}; \boldsymbol{a}_{1,0}^{(\rho)}; \cdots; \boldsymbol{a}_{7,4}^{(\rho)} \right)$$

where $\boldsymbol{a}_{\mu,\nu}^{(\rho)}$ is the concatenated column vector of magnitude matrix $M_{\mu,\nu}^{(\rho)}$ down-sampled by a factor of ρ.

4.3 Gabor-Feature based Robust Representation and Classification

4.3.1 Gabor-feature based robust representation (GRR)

For a testing sample $\boldsymbol{y}_0 \in \Re^m$, the coding model of SRC without occlusion is

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \left\| \boldsymbol{y}_0 - \boldsymbol{A}\boldsymbol{\alpha} \right\|_2^2 + \lambda \left\| \boldsymbol{\alpha} \right\|_1 \right\} \tag{4-2}$$

where $\boldsymbol{A}=[\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K]$ be the concatenation of the n training samples from all the K classes, $\boldsymbol{A}_i = [\boldsymbol{s}_{i,1}, \boldsymbol{s}_{i,2}, ..., \boldsymbol{s}_{i,n_i}] \in \Re^{m \times n_i}$, and $\boldsymbol{s}_{i,j}$, $j=1,2,\ldots,n_i$, is an m-dimensional vector stretched by the j^{th} sample of the i^{th} class. In SRC with occlusion or corruption, the testing sample $\boldsymbol{y}$ is rewritten as

$$\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{e}_0 = \boldsymbol{A}\boldsymbol{\alpha} + \boldsymbol{e}_0 = \boldsymbol{B}\left[\boldsymbol{\alpha}; \boldsymbol{\alpha}_e\right] \tag{4-3}$$

where $\boldsymbol{B} = \left[\boldsymbol{A}, \boldsymbol{A}_e\right] \in \Re^{m \times (n+n_e)}$, and the clean face image $\boldsymbol{y}_0$ and the corruption error $\boldsymbol{e}_0$ have sparse representations over the training sample dictionary $\boldsymbol{A}$ and occlusion dictionary $\boldsymbol{A}_e \in \Re^{m \times n_e}$, respectively.

Images from the same face, taken at (nearly) the same pose but under varying illumination, often lie in a low-dimensional linear subspace known as the *harmonic plane* or *illumination cone* [206, 225]. This implies that if there are only variations of

illumination, SRC can work very well. However, SRC with the holistic image features (e.g., Eigenface [57], Randomface [102] and Fisherface [58] in non-occluded case, or raw pixel intensity value in occluded case) is less effective when there are local deformations of face images, such as certain amount of variations of expressions and pose.

As band-pass filters, Gabor filters could remove some disturbance in the original images and extract more discrimnant features by transforming some information from spatial domain to scale and orientation space. Therefore, the augmented Gabor face feature vector $\boldsymbol{\chi}$ can not only enhance the face feature but also tolerate image local deformation to some extent. So we propose to use $\boldsymbol{\chi}$ to replace the holistic face features for face representation, and the Gabor-feature based representation without face occlusion is

$$\boldsymbol{\chi}(\boldsymbol{y}_0) = \boldsymbol{\chi}(\boldsymbol{A}_1)\boldsymbol{\beta}_1 + \boldsymbol{\chi}(\boldsymbol{A}_2)\boldsymbol{\beta}_2 + \cdots + \boldsymbol{\chi}(\boldsymbol{A}_K)\boldsymbol{\beta}_K = \boldsymbol{\chi}(\boldsymbol{A})\boldsymbol{\beta} \tag{4-4}$$

where $\boldsymbol{\chi}(\boldsymbol{A}) = \left[\boldsymbol{\chi}(\boldsymbol{A}_1), \boldsymbol{\chi}(\boldsymbol{A}_2), \cdots, \boldsymbol{\chi}(\boldsymbol{A}_K)\right]$, $\boldsymbol{\chi}(\boldsymbol{A}_i) = \left[\boldsymbol{\chi}(\boldsymbol{s}_{i,1}), \boldsymbol{\chi}(\boldsymbol{s}_{i,2}), \cdots, \boldsymbol{\chi}(\boldsymbol{s}_{i,n_i})\right]$, $\boldsymbol{\beta} = [\boldsymbol{\beta}_1; \boldsymbol{\beta}_2; \cdots; \boldsymbol{\beta}_K]$, and $\boldsymbol{\chi}(\boldsymbol{s}_{i,j})$ is the augmented Gabor feature vector of $\boldsymbol{s}_{i,j}$.

When the testing face image is occluded, similar to SRC, an occlusion dictionary with Gabor features could be introduced to code the occlusion components, and the Gabor-feature based robust representation could be formulated as:

$$\boldsymbol{\chi}(\boldsymbol{y}) = \left[\boldsymbol{\chi}(\boldsymbol{A}), \boldsymbol{\chi}(\boldsymbol{A}_e)\right][\boldsymbol{\beta}; \boldsymbol{\beta}_e] = \boldsymbol{\chi}(\boldsymbol{B})\boldsymbol{\omega} \tag{4-5}$$

where $\boldsymbol{\chi}(\boldsymbol{A}_e)$ is the Gabor-feature based occlusion dictionary, and $\boldsymbol{\beta}_e$ is the coding vector of the input Gabor feature vector $\boldsymbol{\chi}(\boldsymbol{y})$ over $\boldsymbol{\chi}(\boldsymbol{A}_e)$.

For the convenience of expression, we call the representation in either Eq. (4-4) (for FR without occlusion) or Eq. (4-5) (for FR with occlusion) the Gabor-feature based robust representation (GRR), and the representation vector in the GRR model can be solved by

$$\min_{\boldsymbol{\beta}} \left\{ \left\| \boldsymbol{\chi}(\boldsymbol{y}_0) - \boldsymbol{\chi}(\boldsymbol{A})\boldsymbol{\beta} \right\|_2^2 + \lambda \left\| \boldsymbol{\beta} \right\|_{l_p} \right\} \text{ or } \min_{\boldsymbol{\omega}} \left\{ \left\| \boldsymbol{\chi}(\boldsymbol{y}) - \boldsymbol{\chi}(\boldsymbol{B})\boldsymbol{\omega} \right\|_2^2 + \lambda \left\| \boldsymbol{\omega} \right\|_{l_p} \right\} \tag{4-6}$$

where $\left\| \bullet \right\|_{l_p}$ means the l_p-norm, and p=1 or 2 in this Chapter. In the case of occlusion, the selection of occlusion dictionary $\boldsymbol{\chi}(\boldsymbol{A}_e)$ has a big affect on the performance of GRR, and

thus one key issue is how to define $\chi(\boldsymbol{A}_e)$ to make the GRR effective and efficient.

4.3.2 Discussions on occlusion dictionary

SRC [102] is successful in solving the problem of face occlusion by introducing an occlusion dictionary $\boldsymbol{A}_e$ to code the occluded face components; however, one drawback of SRC is that the number of atoms in the used occlusion dictionary is very big. More specifically, the identity matrix was employed in SRC so that the number of atoms equals to the dimensionality of the image feature vector. For example, if the feature vector has a dimensionality of 3000, then the occlusion dictionary is of size 3000×3000. Such a high dimensional dictionary makes the sparse coding very expensive, and even computationally prohibitive. Suppose the size of the dictionary is $m \times n$, then the empirical complexity of the commonly used l_1-regularized sparse coding methods (such as l_1_ls [204], l_1_magic [226], and MOSEK [227]) to solve Eq. (4-2) is $O(m^2 n^{\varepsilon})$ with $\varepsilon \approx 1.5$ [204, 228]. So if the number of atoms (i.e., n) in the occlusion dictionary is too big, the computational cost will be huge, especially in dealing with FR with occlusion.

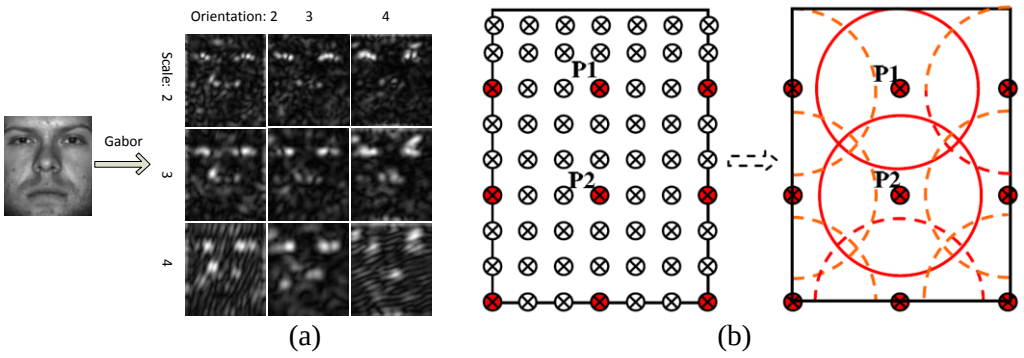


Figure 4.1: Gabor feature extraction. (a) Multi-scale and multi-orientation Gabor filtering; (b) The uniform down-sampling of Gabor feature extraction after Gabor filtering.

By using Gabor features for face representation, the feature dictionary $\boldsymbol{A}$ and the occlusion dictionary $\boldsymbol{A}_e$ in Eq. (4-3) will be transformed into the Gabor feature dictionary

$\chi(\boldsymbol{A})$ and the Gabor-feature based occlusion dictionary $\chi(\boldsymbol{A}_e)$ in Eq. (4-5). Fortunately, $\chi(\boldsymbol{A}_e)$ is compressible. This can be easily illustrated by Fig. 4.1.

Fig. 4.1(a) illustrates the process of Gabor filtering. It is easy to see there are a rich amount of redundancies in the filtering responses across the spatial domain and different scales and orientations. Therefore after the band-pass Gabor filtering of the face images, a uniform spatial down-sampling with a factor of ρ is conducted to form the augmented Gabor feature vector $\boldsymbol{\chi}$, as indicated by the red pixels in Fig. 4.1(b). The spatial down-sampling is performed for all the Gabor filtering outputs along different orientations and on different scales. Therefore, the number of (spatial) pixels in the augmented Gabor feature vector $\boldsymbol{\chi}$ is $1/\rho$ times that of the original face image; meanwhile, at each location, e.g., P1 or P2 in Fig. 1(b), there is a set of directional and scale features extracted by Gabor filtering in the neighborhood (e.g., the circles centered on P1 and P2). Certainly, the directional and scale features at the same spatial location have some correlation, and there are often some overlaps between the supports of Gabor filters, which make the Gabor features at neighboring positions also have some redundancies.

Considering that "occlusion" is a phenomenon of spatial domain, a spatial down-sampling of the Gabor features with a factor of ρ implies that we can use approximately $1/\rho$ times the occlusion bases to code the Gabor features of the occluded face image. In other words, the Gabor-feature based occlusion dictionary $\chi(\boldsymbol{A}_e)$ can be compressed because the Gabor features are redundant as we discussed above. To validate this conclusion, we suppose that the image size is 50×50, and in the original SRC the occlusion dictionary is an identity matrix $\boldsymbol{A}_e = \boldsymbol{I} \in \Re^{2500\times 2500}$. Then the Gabor-feature based occlusion matrix $\chi(\boldsymbol{A}_e) \in \Re^{2560\times 2500}$, where the dimensionality of augmented Gabor feature is 2560 with ρ=39.06, μ={0,…,7}, v={0,…,4}. Fig. 4.2 shows the singular values of $\chi(\boldsymbol{A}_e)$. Obviously, although all the basis vectors of identity matrix $\boldsymbol{I}$ (i.e., $\boldsymbol{A}_e$) have equal importance, only a few (60, with energy proportion of 99.67%) singular vectors of $\chi(\boldsymbol{A}_e)$ have significant singular values, as shown in Fig. 4.2. This implies that $\chi(\boldsymbol{A}_e)$ can be much

more compactly represented by using only a few atoms generated from $\chi(\boldsymbol{A}_e)$, often with a compression ratio about ρ:1. For example, in this experiment we have 2500/60=41.7≈ρ=39.06. Next we present an algorithm to compute a more compact occlusion dictionary.

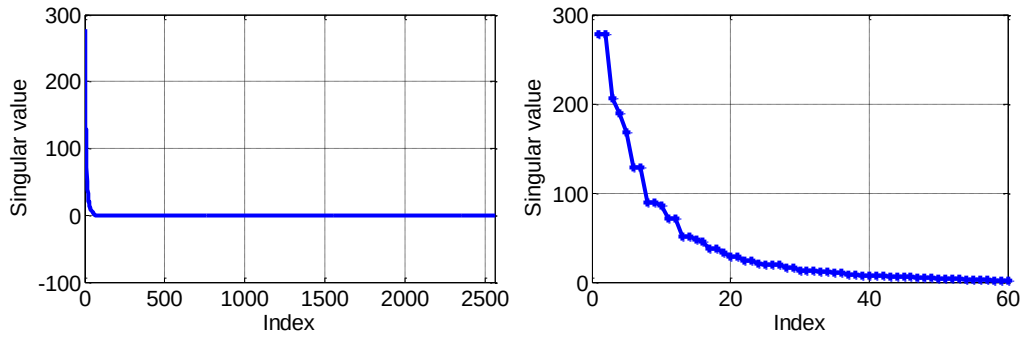


Figure 4.2: The singular values (left: all the singular values, right: the first 60 singular values) of Gabor feature-based occlusion matrix.

4.3.3 Gabor occlusion dictionary (GOD) computing

Now that $\chi(\boldsymbol{A}_e)$ is compressible, we propose to compute a compact occlusion dictionary from it with suitable regularization on the coefficients. Here a compact dictionary, denoted by $\boldsymbol{D} \in \Re^{m \times n}$, refers to a dictionary which has much less columns (i.e., the so-called atoms) than rows (i.e., n<<m). We call the computed compact occlusion dictionary the Gabor occlusion dictionary (GOD) and denote it as $\boldsymbol{\Gamma}$. Then we could replace $\chi(\boldsymbol{A}_e)$ by $\boldsymbol{\Gamma}$ in the GRR based FR.

For the convenience of expression, we denote by $\boldsymbol{Z} = \chi(\boldsymbol{A}_e) = [\boldsymbol{z}_1, \cdots, \boldsymbol{z}_{n_e}] \in \Re^{m_\rho \times n_e}$ the original Gabor-feature based occlusion matrix, with each column $\boldsymbol{z}_i$ being the augmented Gabor-feature vector generated from each atom of $\boldsymbol{A}_e$. The compact occlusion dictionary to be computed is denoted by $\boldsymbol{\Gamma} = [\boldsymbol{d}_1, \boldsymbol{d}_2, ..., \boldsymbol{d}_q] \in \Re^{m_\rho \times q}$, where q can be set as slightly less than n_e/ρ in practice. It is required that each occlusion basis $\boldsymbol{d}_j$, j=1, 2, …, q, is a unit column vector, i.e. $\boldsymbol{d}_j^T \boldsymbol{d}_j = 1$. Since we want to replace $\boldsymbol{Z}$ by $\boldsymbol{\Gamma}$, it is expected that

the original dictionary $\boldsymbol{Z}$ can be well represented by $\boldsymbol{\Gamma}$ with the representation coefficients being regularized via l_p-norm regularization. Obviously, p=1 means that we require sparse representation on the learnt GOD. Inspired by the success of l_2-norm regularization in CRC [179-180] (or please refer to Chapter 3), we can also use l_2-norm coefficient regularization. With such considerations, the objective function for determining $\boldsymbol{\Gamma}$ is defined as:

$$\min_{\Gamma,\Lambda}\left\{\left\|\boldsymbol{Z}-\boldsymbol{\Gamma}\boldsymbol{\Lambda}\right\|_F^2+\zeta\left\|\boldsymbol{\Lambda}\right\|_{l_p}\right\}\ \text{ s.t. }\ \boldsymbol{d}_j^T\boldsymbol{d}_j=1,\forall j \tag{4-7}$$

where $\boldsymbol{\Lambda}$ is the representation matrix of $\boldsymbol{Z}$ over dictionary $\boldsymbol{\Gamma}$, ζ is a positive scalar that balances the F-norm term and the l_p-norm term (here p=1 for $\|\cdot\|_1$ and p=2 for $\|\cdot\|_F^2$).

Eq. (4-7) is a joint optimization problem of the occlusion dictionary $\boldsymbol{\Gamma}$ and the representation matrix $\boldsymbol{\Lambda}$. Like in many multi-variable optimization problems, we solve Eq. (4-7) by optimizing $\boldsymbol{\Gamma}$ and $\boldsymbol{\Lambda}$ alternatively. The optimization procedures are described in Table 4.2.

Table 4.2: Algorithm of Gabor occlusion dictionary computing.

Algorithm of Gabor occlusion dictionary (GOD) computing
1. Initialize $\boldsymbol{\Gamma}$ We initialize each column of $\boldsymbol{\Gamma}$ as a random vector with unit l_2-norm. **2. Fix $\boldsymbol{\Gamma}$ and solve $\boldsymbol{\Lambda}$** By fixing $\boldsymbol{\Gamma}$, the objective function in Eq. (4-7) will be reduced to $$\min_{\Lambda}\left\{\left\|\boldsymbol{Z}-\boldsymbol{\Gamma}\boldsymbol{\Lambda}\right\|_F^2+\zeta\left\|\boldsymbol{\Lambda}\right\|_{l_p}\right\} \tag{4-8}$$ The minimization of Eq. (4-8) for p=1 can be achieved by the l_1-norm minimization techniques. In this paper, we use the algorithm in [204]. The minimization of Eq. (4-8) for p=2 could be efficiently solved since has a closed-form least square solution [180]. **3. Fix $\boldsymbol{\Lambda}$ and update $\boldsymbol{\Gamma}$** Now the objective function is reduced to $$\min_{\Gamma}\left\{\left\|\boldsymbol{Z}-\boldsymbol{\Gamma}\boldsymbol{\Lambda}\right\|_F^2\right\}\ \text{ s.t. }\ \boldsymbol{d}_j^T\boldsymbol{d}_j=1,\forall j \tag{4-9}$$ We can write matrix $\boldsymbol{\Lambda}$ as $\boldsymbol{\Lambda}=[\boldsymbol{\beta}_1;\boldsymbol{\beta}_2,\ldots,\boldsymbol{\beta}_q]$, where $\boldsymbol{\beta}_j$, j=1,2,…,q, is the row vector of $\boldsymbol{\Lambda}$. We update the occlusion bases one by one. When updating $\boldsymbol{d}_j$, all the other columns of $\boldsymbol{\Gamma}$, i.e., $\boldsymbol{d}_l$, $l\neq j$, are

fixed. Then Eq. (4-9) is converted into

$$\min_{\boldsymbol{d}_j} \left\| \boldsymbol{Z} - \sum_{l \neq j} \boldsymbol{d}_l \boldsymbol{\beta}_l - \boldsymbol{d}_j \boldsymbol{\beta}_j \right\|_F^2 \quad \text{s.t.} \quad \boldsymbol{d}_j^T \boldsymbol{d}_j = 1 \tag{4-10}$$

Let $\boldsymbol{Y} = \boldsymbol{Z} - \sum_{l \neq j} \boldsymbol{d}_l \boldsymbol{\beta}_l$, Eq. (4-10) can be written as

$$\min_{\boldsymbol{d}_j} \left\| \boldsymbol{Y} - \boldsymbol{d}_j \boldsymbol{\beta}_j \right\|_F^2 \quad \text{s.t.} \quad \boldsymbol{d}_j^T \boldsymbol{d}_j = 1 \tag{4-11}$$

Using Langrage multiplier, Eq. (4-10) is equivalent to

$$\min_{\boldsymbol{d}_j} tr\left(-\boldsymbol{Y}\boldsymbol{\beta}_j^T \boldsymbol{d}_j^T - \boldsymbol{d}_j \cdot \boldsymbol{\beta}_j \boldsymbol{Y}^T + \boldsymbol{d}_j \cdot (\boldsymbol{\beta}_j \boldsymbol{\beta}_j^T - \gamma) \boldsymbol{d}_j^T + \gamma \right) \tag{4-12}$$

where γ is a scalar variable. Differentiating Eq. (4-12) with respect to $\boldsymbol{d}_j$, and let it be $\boldsymbol{0}$, we have

$$\boldsymbol{d}_j = \boldsymbol{Y}\boldsymbol{\beta}_j^T \left(\boldsymbol{\beta}_j \boldsymbol{\beta}_j^T - \gamma \right)^{-1} \tag{4-13}$$

Since $\left(\boldsymbol{\beta}_j \boldsymbol{\beta}_j^T - \gamma \right)$ is a scalar and γ is a variable, the solution of Eq. (4-13) under constrain $\boldsymbol{d}_j^T \boldsymbol{d}_j = 1$ is

$$\boldsymbol{d}_j = \boldsymbol{Y}\boldsymbol{\beta}_j^T \Big/ \left\| \boldsymbol{Y}\boldsymbol{\beta}_j^T \right\|_2 \tag{4-14}$$

Using the above procedures, we can update all the vectors $\boldsymbol{d}_j$, and hence the whole set $\boldsymbol{\Gamma}$ is updated.

4. Output $\boldsymbol{\Gamma}$

Go back to step 2 until the object function values in adjacent iterations are close enough, or the maximum number of iterations is reached. Finally, output $\boldsymbol{\Gamma}$.

It is straightforward that the above GOD computing algorithm converges because in each iteration $J_{\boldsymbol{\Gamma},\boldsymbol{A}}$ will decrease, as illustrated in Fig. 4.3. Consequently, in our proposed GRR, we use the GOD $\boldsymbol{\Gamma}$ to replace the $\boldsymbol{\chi}(\boldsymbol{A}_e)$ in Eq. (4-5). Finally, the coding problem in GRRC with face occlusion is

$$\min_{\boldsymbol{\omega}} \left\{ \left\| \boldsymbol{\chi}(\boldsymbol{y}) - \boldsymbol{B}_{\boldsymbol{\Gamma}} \boldsymbol{\omega}_{\boldsymbol{\Gamma}} \right\|_2^2 + \lambda \left\| \boldsymbol{\omega}_{\boldsymbol{\Gamma}} \right\|_{l_p} \right\} \text{ where } \boldsymbol{B}_{\boldsymbol{\Gamma}} = \left[\boldsymbol{\chi}(\boldsymbol{A})\, \boldsymbol{\Gamma} \right], \boldsymbol{\omega}_{\boldsymbol{\Gamma}} = \left[\boldsymbol{\beta}; \boldsymbol{\beta}_{\boldsymbol{\Gamma}} \right] \tag{4-15}$$

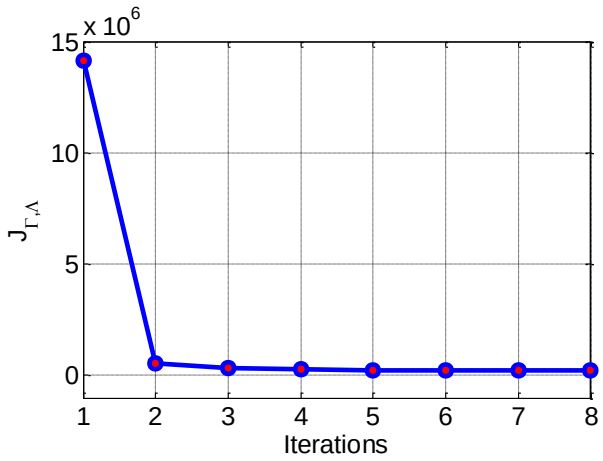


Figure 4.3: Illustration of the convergence of the proposed Gabor occlusion dictionary (GOD) computing algorithm on AR database. A GOD with 100 atoms is computed from the original Gabor-feature based occlusion matrix with 4980 columns. The compression ratio is nearly 50:1.

4.3.4 GRR based classification (GRRC)

The SRC scheme [102] assumes that the face image representation residual is sparse, and thus uses the l_1-norm to characterize the representation coefficients associated with the occlusion dictionary, i.e., the identity matrix. Because the number of atoms in the identity matrix is very big (equal to the dimensionality of face image), it is necessary to impose the l_1-norm sparsity on the coding coefficients for a robust and unique representation, yet this makes the complexity of SRC very high. However, when Gabor feature is adopted, a compact GOD $\boldsymbol{\Gamma}$ (with only about 1/40 times the size of the identity matrix) can be learnt, and thus it may not be necessary to use the l_1-norm sparsity to regularize the coding coefficients over the dictionary anymore.

For a given face Gabor feature $\boldsymbol{\chi}(\boldsymbol{y})$, often its dimensionality is much higher than the number of atoms in dictionary $\boldsymbol{B}_{\boldsymbol{\Gamma}}=[\boldsymbol{\chi}(\boldsymbol{A})\ \boldsymbol{\Gamma}]$ after GOD computing, which means that the dictionary $\boldsymbol{B}_{\boldsymbol{\Gamma}}$ is not over-complete, and hence the system

$$\boldsymbol{\chi}(\boldsymbol{y}) \approx \boldsymbol{B}_{\boldsymbol{\Gamma}}\boldsymbol{\omega}_{\boldsymbol{\Gamma}} \tag{4-16}$$

is generally an over-determined system. This implies that the solution of Eq. (4-16) is stable even without any regularization, and thus it is not necessary to require the

representation coefficient vector $\boldsymbol{\omega}_{\boldsymbol{\Gamma}}$ to be sparse in order for an accurate approximation of $\boldsymbol{\chi}(\boldsymbol{y})$ by $\boldsymbol{B}_{\boldsymbol{\Gamma}}$. With these considerations, in this Chapter we allow $\boldsymbol{\omega}_{\boldsymbol{\Gamma}}$ to be either sparse or dense, and test the results by using both l_1-norm and l_2-norm to regularize the coding coefficients. We name the GRR based classification (GRRC) with l_1-norm regularization GRRC_L$_1$, and the GRRC with l_2-norm regularization GRRC_L$_2$. The GRRC algorithm is summarized in Table 4.3.

Table 4.3: Algorithm of GRR based Classification (GRRC).

Algorithm of GRRC
1. **Input:** Gabor feature dictionary $\boldsymbol{\chi}(\boldsymbol{A})$, GOD $\boldsymbol{\Gamma}$, and the Gabor feature $\boldsymbol{\chi}(\boldsymbol{y}_o)$ (for testing sample without occlusion) or $\boldsymbol{\chi}(\boldsymbol{y})$ (for testing sample with occlusion).
2. Solve the l_p-minimization (p=1 or 2) problem (the Lagrange formulation): $$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \left\| \boldsymbol{\chi}(\boldsymbol{y}_o) - \boldsymbol{\chi}(\boldsymbol{A})\boldsymbol{\beta} \right\|_2^2 + \lambda \left\| \boldsymbol{\beta} \right\|_{l_p} \right\} \quad (4\text{-}17)$$ or (let $\boldsymbol{\omega}_{\Gamma} = [\boldsymbol{\beta}; \boldsymbol{\beta}_{\Gamma}]$) $$\hat{\boldsymbol{\omega}}_{\Gamma} = \arg\min_{\boldsymbol{\omega}_{\Gamma}} \left\{ \left\| \boldsymbol{\chi}(\boldsymbol{y}) - \left[\boldsymbol{\chi}(\boldsymbol{A})\ \boldsymbol{\Gamma} \right] \boldsymbol{\omega}_{\Gamma} \right\|_2^2 + \lambda \left\| \boldsymbol{\omega}_{\Gamma} \right\|_{l_p} \right\} \quad (4\text{-}18)$$ where $\hat{\boldsymbol{\omega}}_{\Gamma} = [\hat{\boldsymbol{\beta}}; \hat{\boldsymbol{\beta}}_{\Gamma}]$ and λ is a positive scalar that balances the coding residual and regularization strength.
3. Compute the residuals $$r_i(\boldsymbol{y}_o) = \left\| \boldsymbol{\chi}(\boldsymbol{y}_o) - \boldsymbol{\chi}(\boldsymbol{A}_i)\hat{\boldsymbol{\beta}}_i \right\|_2, \text{ for } i = 1, \cdots, K. \quad (4\text{-}19)$$ or $$r_i(\boldsymbol{y}) = \left\| \boldsymbol{\chi}(\boldsymbol{y}) - \boldsymbol{\Gamma}\hat{\boldsymbol{\beta}}_{\Gamma} - \boldsymbol{\chi}(\boldsymbol{A}_i)\hat{\boldsymbol{\beta}}_i \right\|_2, \text{ for } i = 1, \cdots, K. \quad (4\text{-}20)$$ where $\hat{\boldsymbol{\beta}}_i$ is the coding coefficient sub-vector associated to class i.
4. **Output:** identity($\boldsymbol{y}_o$)=argmin$_i$ $r_i(\boldsymbol{y}_o)$ or identity ($\boldsymbol{y}$)=argmin$_i$ $r_i(\boldsymbol{y})$.

4.3.5 Time complexity

The empirical complexity of the commonly used l_1-regularized sparse coding methods is $O(m^2 n^{\varepsilon})$ with $\varepsilon \approx 1.5$ [204, 228], while the time complexity of l_2-norm regularized coding is only $O(mn)$ [180] for that the coding projection matrix could be computed offline,

where m is facial feature dimensionality and n is the number of dictionary atoms. For GRRC, in Fourier domain it is very fast to extract Gabor features, whose time complexity could be negligible compared with that of l_1-norm regularized sparse coding.

In the case of FR without occlusion, n is the number of training samples. Therefore, GRRC_L_1 has similar computational burden to SRC, but GRRC_L_2 has much lower time complexity than GRRC_L_1 and SRC. For FR without occlusion, there is a fast version of SRC, namely SRC using Hashing [198]. This method is usually faster than the original SRC because the used random projection matrix is very sparse. So GRRC_L_1 would have a little higher time complexity than SRC using Hashing, but GRRC_L_2 is still much faster than SRC using Hashing.

In the case of FR with occlusion, it is easy to get that the time complexity of GRRC_L_1 is $O(m^2(n+m/\rho)^\varepsilon)$, where $\rho \approx 40$. This is much lower than SRC whose time complexity is $O(m^2(n+m)^\varepsilon)$. Obviously, GRRC_L_2's time complexity is $O(m(n+m/\rho))$ and it is the fastest one among the three methods.

4.4 Experimental Results

In this section, we present experiments on benchmark face databases to demonstrate the superiority of GRRC to SRC. Before giving the detailed experimental results, we discuss the selection of Gabor features and regularization of GOD computing in Section 4.4.1. To evaluate more comprehensively the performance of GRRC, in Section 4.4.2 we first test FR with little deformation; then in Section 4.4.3 we demonstrate the robustness of GRRC to expression and pose variation; finally in Section 4.4.4 we test FR against block occlusion and real disguise. In our implementation of Gabor filters, the parameters are set as $K_{max}=\pi/2$, $f=\sqrt{2}$, $\sigma=1.5\pi$, $\mu=\{0,\ldots,7\}$, $v=\{0,\ldots,4\}$ by our experiences and they are fixed for all the experiments. In the experiments, λ in GRRC is fixed 0.0005 for FR without and with occlusion. We also give the results of GRRC with $\lambda=0.001$ for FR

without occlusion to show GRRC very robust to parameter's value. In addition, all the face images are cropped and aligned by using the location of eyes, which is provided by the face databases (for Mulit-PIE, we manually locates the positions of eyes).

In the following tables of this section, the results of competing methods with reference numbers in the tables are reported by the corresponding paper. All the other results are computed by us with reporting their best recognition rates.

4.4.1 Gabor features and regularization of GOD computing

1) Gabor features: In GRRC, we adopt the Gabor magnitude as the augmented facial features. Here we also evaluate other Gabor features, such as Gabor real parts, Gabor imaginary parts, and the concatenation of Gabor real and imaginary parts. We replace Gabor magnitude features in GRRC_L_2 by these Gabor features, and test their performance on the AR database (the detailed experimental setting is described in Section 4.4.2). Table 4.4 lists the recognition rates. It is easy to see that the features of Gabor real parts (denoted by GRRC_L_2 (Real parts)), Gabor imaginary parts (denoted by GRRC_L_2 (Imaginary parts)) and their concatenation (denoted by GRRC_L_2 (Real + Imaginary)) do not lead to good results. This demonstrates that Gabor magnitude (denoted by GRRC_L_2 (Magnitude)) is more discriminative in the Gabor feature-based representation scheme. The results by SRC [102] and CRC [180] schemes with holistic PCA features are also listed in Table 4.4 for comparison.

Table 4.4: Face recognition rates (%) of different Gabor features on AR database.

Dimension	130	300	540
PCA+SRC	89.7	93.3	93.5
PCA+CRC	90.0	93.7	93.9
GRRC_L_2 (Real parts)	84.3	89.4	91.4
GRRC_L_2 (Imaginary parts)	85.8	91.0	93.3
GRRC_L_2 (Real + Imaginary)	85.0	91.4	93.6
GRRC_L_2 (Magnitude)	**93.1**	**96.8**	**97.3**

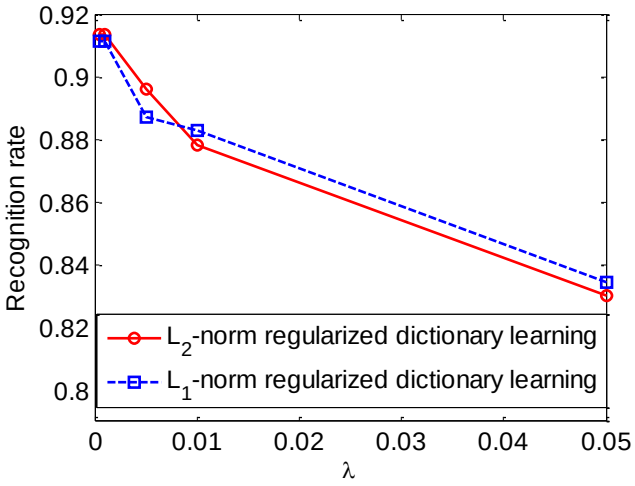


Figure 4.4: Recognition rates by using l_1-norm and l_2-norm regularized GOD computing in the experiment of FR with random block occlusion.

2) Regularization on GOD computing: In the GOD computing algorithm (refer to Table 4.1), we regularize the coding coefficient by l_p-norm with p=1 or 2. Here we use an FR experiment on Extended Yale B [99, 206] with random block face occlusion (about 45% occlusion) to discuss the selection of l_p-norm. The detailed experimental setting will be presented in the experiments of *FR with random block occlusion* in Section 4.4.4. We set the parameter ζ in the model (Eq. (4-7)) of GOD computing as 0.005. The recognition rates of l_p-norm regularized GOD computing versus different regularization parameters λ in coding (Eq. (4-18) with l_1-norm regularization) of the classification stage are shown in Fig. 4.4. It can be seen that there is not much difference in recognition accuracy between l_1-norm and l_2-norm regularization in GOD computing. The reason is that the redundancy of Gabor feature transformation (analyzed in Section 4.3.2) makes the learnt GOD dictionary compact so that the GOD dictionary is obviously over-determined. An over-determined dictionary itself could stably represent the testing sample even without regularization. Therefore, the l_1-norm and l_2-norm regularizations will lead to stable occluded face representation and similar recognition results. Considering that the recognition rates by l_1-norm and l_2-norm regularized GOD computing are similar, we prefer to use the l_2-norm regularized one for its fast speed. In our paper, the parameter ζ in

GOD computing is set as a small scalar, e.g., 0.001.

In order to give an intuitive illustration of the leant GOD, we plot the 1st, 51st, 101st and 151st atom of l_1-norm regularized GOD in Fig. 4.5. We could see that the learnt GOD atoms are roughly periodic signals, which have 40 repeated patterns (because the Gabor feature is the concatenation of 40 down-sampled Gabor magnitudes). The original occlusion dictionary (i.e., the identity matrix) has clear spatial meaning, e.g., each atom is a unit vector representing one pixel of the image. However, the size of such an occlusion dictionary is too big (e.g., 8064×8064 in this experiment). The learnt GOD not only has much smaller size (e.g., 8940×200), but also have very clear spatial meaning, i.e., on each down-sampled Gabor magnitude feature, the corresponding atom of GOD is a local basis to represent the scale and orientation information at that location. Therefore, GOD is much more efficient to handle occlusion.

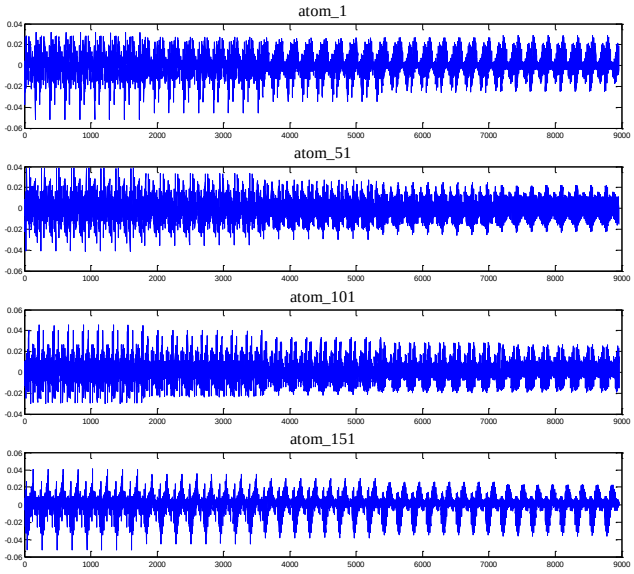


Figure 4.5: The 1st, 51st, 101st, and 151st atoms of the learnt Gabor Occlusion Dictionary.

4.4.2 Face recognition with little deformation

We evaluate the proposed GRRC scheme on four representative facial image databases:

Extended Yale B [99, 206], AR [212], Multi-PIE [213] and FERET [45, 229]. We compare GRRC with SRC [102], CRC [180], Linear Regression for Classification (LRC) [101], linear Support Vector Machine (SVM) and Nearest Neighbor (NN) methods. If no specific instruction, for all the competing methods we use PCA to reduce the feature dimension.

1) Extended Yale B Database: The Extended Yale B database consists of 2,414 frontal-face images of 38 individuals, captured under various laboratory-controlled lighting conditions [99, 206]. For each subject, we randomly selected half of the images for training (i.e., 32 images per subject), and used the other half for testing. The images are normalized to 192×168, and the dimension of the augmented Gabor feature vector of each image is 19760 (40×26×19). The results of all the methods versus the feature dimension are listed in Table 4.5. It can be seen that GRRC is better than SRC, CRC and other methods in all the dimensions except that SRC and LRC are slightly better GRRC_L_2 in the dimension of 56. GRRC_L_2 has similar performance to GRRC_L_1 when the dimension is greater than 56. On this database, the maximal recognition rates of the competing methods are 99.2% for GRRC_L_1, 99.1% for GRRC_L_2, 97.9% for SRC, 98.0% for CRC, 96.4 for SVM, 95.7% for LRC, and 92.0% for NN. In addition, it can be seen that GRRC is not sensitive to the value of λ.

Table 4.5: Face recognition results (%) on the Extended Yale B database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

Methods	56	120	300	504
SRC	92.6	95.6	97.4	97.9
CRC	88.6	95.4	97.4	98.0
NN	81.4	89.2	91.9	92.0
LRC	94.1	94.7	95.4	95.7
SVM	92.6	95.3	96.3	96.4
GRRC_L_1	**92.7(92.7)**	95.6(96.2)	97.9(97.9)	99.0**(99.2)**
GRRC_L_2	90.5(90.5)	**96.3(96.3)**	**98.4(98.4)**	99.1(99.1)

2) AR database: The AR database consists of over 4,000 frontal images from 126 individuals [212]. For each individual, 26 pictures were taken in two separate sessions. As

in [102], in the experiment we chose a subset of the dataset consisting of 50 male subjects and 50 female subjects. For each subject, the seven images with illumination change and expressions from Session 1 were used for training, and the other seven images with only illumination change and expression from Session 2 were used for testing. The size of original face image is 165×120, and the Gabor-feature vector is of dimension 12000 (40×20×15). The comparison of GRRC and the competitors are shown in Table 4.6. Again we can see that GRRC performs much better than all the other methods under all the dimensions, especially with more than 3% improvement when the dimension is lager than 54. On this database, the maximal recognition rate of GRRC_L_1, GRRC_L_2, SRC, CRC, SVM are 97.1%, 97.3%, 93.5%, 93.9% and 88.8%, respectively.

Table 4.6: Face recognition results (%) on the AR database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

Methods	54	130	300	540
SRC	80.0	89.7	93.3	93.5
CRC	80.5	90.0	93.7	93.9
NN	67.8	70.1	71.2	72.1
LRC	75.4	76.0	70.7	76.7
SVM	77.5	82.7	87.3	88.8
GRRC_L_1	**86.0(86.0)**	**94.0(94.0)**	**96.7**(96.6)	97.1(97.1)
GRRC_L_2	82.7(82.7)	93.1(93.1)	**96.7(96.7)**	**97.3(97.3)**

3) Large-scale Multi-PIE database: The CMU Multi-PIE database [213] 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 illuminations {0,1,3,4,6,7,8,11,13,14,16,17,18,19} and neutral expression. For the testing sets, 10 typical frontal images of even-number illuminations taken with neutral expressions from Session 2 to Session 4 were used. The image size is cropped and normalized to 100×82, and the Gabor feature vector is of the dimension of 8320 (40×16×13). We use PCA to reduce the dimensionality of the input feature to 300. Table 4.7 lists the recognition rates

in three tests by the competing methods. The results validate that GRRC methods get the best in accuracy, at least 3% higher than that of SRC and CRC in session 2 and about 5% higher than that of SRC and CRC in other sessions. NN, LRC and SVM can not get good recognition accuracy (lower than 90%) in this database, much lower than SRC, CRC and GRRC.

Table 4.7: Face recognition results (%) on the Multi-PIE database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

	SRC	CRC	NN	LRC	SVM	**GRRC_L$_1$**	**GRRC_L$_2$**
Session 2	93.9	94.1	86.4	87.1	85.2	97.3**(97.5)**	97.1(97.2)
Session 3	90.0	89.3	78.8	81.9	78.1	96.7(96.7)	**96.8(96.8)**
Session 4	94.0	93.3	82.3	84.3	82.1	98.6(98.6)	**98.7(98.7)**

Table 4.8: Face recognition results (%) on the FERET database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

	SRC	CRC	NN	SVM	**GRRC_L$_1$**	**GRRC_L$_2$**
Fb	86.9	85.4	87.1	87.1	**95.7**(95.6)	95.6(95.6)
Fc	77.3	75.8	73.2	73.2	**97.4(97.4)**	94.8(95.4)
Dup1	51.6	51.5	47.8	47.8	77.7(78.0)	**79.1**(78.9)
Dup2	33.3	35.5	23.9	23.9	75.6(76.5)	**78.6(78.6)**

4) Large-scale FERET database: The FERET database [45, 229] is often used to validate an algorithm's effectiveness because it contains many kinds of image variations. By taking 'Fa' subset as a gallery, the probe subsets 'Fb' and 'Fc' were captured with expression and illumination variations. Especially, 'Dup1' and 'Dup2' consist of images that were taken at different times with more than one year interval. Here we should note that in the Gallery set 'Fa', each subject only has one sample, which is very challenging for SRC and GRRC because usually they usually need several samples for each subject to construct the subspace. The image size is cropped and normalized to 150×130, and the Gabor feature vector is of dimension 21000 (40×25×21). For all the competing methods, we used LDA to reduce the original feature dimensionality to 428 for LDA could achieve better performance than PCA in this challenging dataset. Table 4.8 shows the face

recognition results on FERET database. It is surprised that SRC and CRC have higher accuracy than NN and SVM except for 'Fb' even only one sample for each subject in the training set. GRRC methods achieve the best performance with over 95% recognition rates in 'Fb' and 'Fc' and about 78% in 'Dup1' and 'Dup2'. It can also be seen that for 'Fb', GRRC has at least 8% improvements compared to other methods, while with about 20%, 27% and 43% improvements for 'Fc', 'Dup1' and 'Dup2', respectively. According to the recent state-of-the-art FR results on the FERET database, e.g., Xie *et al*. 's method [94], further improvement could be achieved if more discriminative features, e.g., fused Gabor magnitude and phase feature [94], are utilized in the framework of GRRC.

From the experimental results in Extended Yale B, AR, Multi-PIE and FERET, we could see that GRRC is very robust to the value of λ and GRRC_L_1 and GRRC_L_2 have very similar performance (the gap usually is less than 0.5% in high dimensional feature), showing that GRRC_L_2 is very suitable for the practical FR systems due to its fast speed and good performance. Besides, the improvements brought by GRRC on the AR, Multi-PIE, and FERET are much bigger than that on the Extended Yale B database. This is because mostly there is only illumination variation between the training images and testing images, and the number of training samples (i.e., 32) in the Extended Yale B database is also high. Thus the original SRC and CRC work well on it. However for the more challenging cases (e.g., the training and testing samples of the AR, Multi-PIE and FERET have much more variations, including time, illumination, etc., but with very limited number of training samples), the local feature based GRRC is much more robust than the holistic feature based SRC, CRC, SVM, LRC and NN.

4.4.3 Face recognition with pose and expression variations

In this section, we verify the robustness of GRRC to pose and expression variations on the pose subset of FERET database [45, 229] and expression subset of Multi-PIE [213].

1) FERET pose database: Here we used the pose subset of the FERET database [45, 229], which includes 1400 images from 198 subjects (about 7 each). This subset is composed of the images marked with '*ba*', '*bd*', '*be*', '*bf*', '*bg*', '*bj*', and '*bk*'. In our experiment, each image has the size of 80×80 and the dimension of Gabor feature is 6760 (40×13×13). Some sample images of one person are shown in Fig. 4.6.

ba: gallery bj: expression bk: illumination be: +15 bf: -15 bg: -25 bd: +25

Figure 4.6: Some samples of a subject on the pose subset of the FERET database.

Five tests with different pose angles were performed. In test 1 (pose angle is zero degree), images marked with '*ba*' and '*bj*' were used as the training set, and images marked with '*bk*' were used as the testing set. In all the other four tests, we used images marked with '*ba*', '*bj*' and '*bk*' as gallery, and used the images with '*bg*', '*bf*', '*be*' and '*bd*' as probes, respectively. Here we use 350-dimension Eigenfaces as the input feature. Table 4.9 lists the results of different methods for various face poses. Obviously, we can see that GRRC has much higher recognition rates than SRC and other methods. In particular, when the pose variation is moderate (0^o and $\pm 15^o$), about 20% improvement is achieved by GRRC compared to SRC. We could also see that GRRC_L_2 performs very similarly to GRRC_L_1. It is undeniable that GRRC's performance also degrades much when pose variation becomes large (e.g. $\pm 25^o$). Nevertheless, GRRC can much improve the robustness to moderate pose variation, which indicating GRRC could tolerate registration error (e.g., pose variation, misalignment) to some extent.

Table 4.9: Face recognition results (%) on the pose subset of the FERET database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

Pose (degree)	-25	-15	0	15	25
SRC	32.5	70.5	83.5	57.5	28.0
CRC	21.0	62.5	74.5	40.0	17.0
NN	10.5	54.0	78.5	39.0	17.5
LRC	11.5	58.0	75.5	40.5	20.0
SVM	14.5	61.5	80.5	43.5	20.0
GRRC_L$_1$	41.5**(42.0)**	**95.5(95.5)**	**99.0(99.0)**	89.0(89.5)	44.5**(45.0)**
GRRC_L$_2$	41.0(41.0)	**95.5(95.5)**	**99.0(99.0)**	**91.5(91.5)**	44.0(44.0)

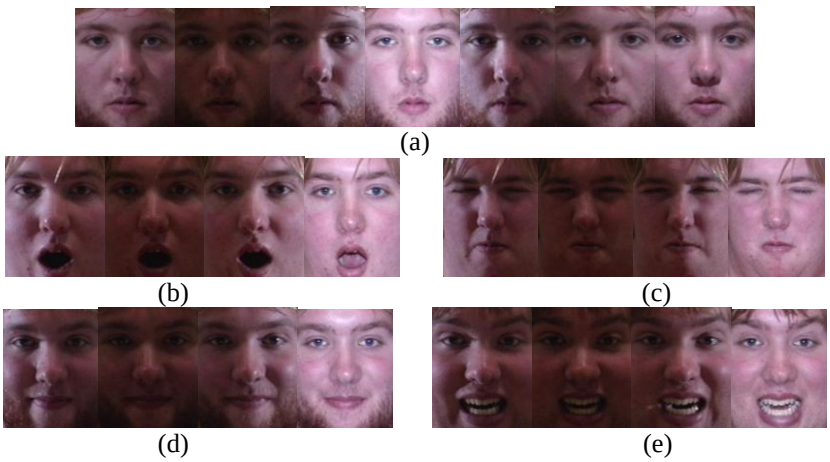


Figure 4.7: A subject in Multi-PIE database. (a) Training samples with only illumination variations. (b) Testing samples with surprise expression and illuminations in Session 2. (c) Testing samples with squint expression and illuminations in Session 2. (d) and (e) show the testing samples with smile expression and illumination variations in Session 1 and Session 3, respectively.

2) Multi-PIE expression subset: All the 249 subjects in Session 1 were used as training set in this experiment. To make the FR more challenging, four subsets with both illumination and expression variations in Sessions 1, 2 and 3 were used for testing. For the training set, as in [189] we used the 7 frontal images with extreme illuminations {0, 1, 7, 13, 14, 16, 18} and neutral expression (refer to Fig. 4.7(a) for examples). For the testing set, 4 typical frontal images with illuminations {0, 2, 7, 13} and different expressions (smile in Sessions 1 and 3, squint and surprise in Session 2) are used (refer to Fig.4.7(b) for examples with surprise in Session 2, Fig. 4.7(c) for examples with squint in Session 2, Fig. 4.7(d) for examples with smile in Session 1, and Fig. 4.7(e) for examples with smile

in Session 3). We used the Eigenface with dimensionality 900 as the face feature.

Table 4.10: Face recognition rates on Multi-PIE expression database. For GRRC, $r_1(r_2)$ means r_1 is the recognition rate for λ=0.0005, with r_2 for λ=0.001.

	Smile-S1	Smile-S3	Surprise-S2	Squint-S2
SRC	94.1	60.9	55.0	57.2
CRC	92.4	56.7	49.2	52.7
NN	89.4	46.3	40.5	50.3
LRC	90.4	49.8	40.1	52.1
SVM	88.9	46.3	25.6	47.7
Hash+OMP	92.2	50.2	42.3	51.8
Hash+L_1	87.2	50.0	46.4	56.2
GRRC_L_1	**97.7**(97.4)	73.4(73.3)	81.8**(82.4)**	87.5(87.8)
GRRC_L_2	97.3(97.3)	**74.2(74.2)**	82.2(82.2)	88.0**(88.1)**

Table 4.10 lists the recognition rates in four testing sets by the competing methods, including SRC using Hasing [198] (e.g., Hash+OMP and Hash+L_1). It can be seen that GRRC achieves the best performance in all tests and SRC performs the second best. It can also be seen that SRC using Hashing has low recognition rates than SRC, which may result from it using random projection dimension-reduced matrix. In addition, all the methods achieve their best results when Smile-S1 is used for testing because the training set is also from Session 1. The highest rates of GRRC_L_1 and GRRC_L_2 are 97.7% and 97.3%, respectively, more than 3% improvement over the third best one, SRC. From testing set Smile-S1 to set Smile-S3, the variations increase because of the longer data acquisition time interval and expression changes (refer to Fig. 4.7 (d) and Fig. 4.7 (e)). The recognition rates of GRRC_L_1 and GRRC_L_2 drop by 24.3% and 23.1%, respectively, while those of SRC, CRC, NN, LRC and SVM drop by 33.2%, 35.7%, 43.1%, 40.6% and 42.6%, respectively, validating that GRRC is much more robust to face variation than the other methods. For the testing set of Surpise-S2 and Squint-S2, GRRC has about 30% improvement over all the other methods. Meanwhile, for all the four tests, GRRC with l_1-norm constraint or l_2-norm constraint on coding coefficients has similar performance.

Table 4.11: Average time (second) comparison for sparse representation-based FR methods

Method	SRC	Hash+OMP	Hash+L_1	GRRC_L_1	GRRC_L_2
Running Time	1.398	1.061	2.644	0.2423+1.400	0.2423+0.046

The running time of GRRC, SRC, and SRC using Hashing [198] (e.g., Hash+OMP and Hash+L_1) is compared in Table 4.11. Here the Gabor feature extraction for GRRC is 0.2423 second. From Table 4.11, it is very clear that GRRC_L_2 is the fastest one, about 4 times faster than Hash+OMP, the second fastest method. GRRC_L_1's running time is still lower than Hash+L_1.

From the experiments on FR with local deformation (e.g., pose and expression variations), we could see that there is almost no difference for GRRC with λ=0.001 and GRRC with λ=0.0005, showing GRRC is very robust to the value of λ. GRRC is much superior to the other methods, including SRC and CRC. This not only shows that collaborative representation based classification strategy with l_1 or l_2 norm regularization is more powerful than other classifiers, such as NN, LRC and SVM, but also demonstrates that Gabor magnitude features are more robust to the variations of pose and expression.

4.4.4 Recognition against occlusion

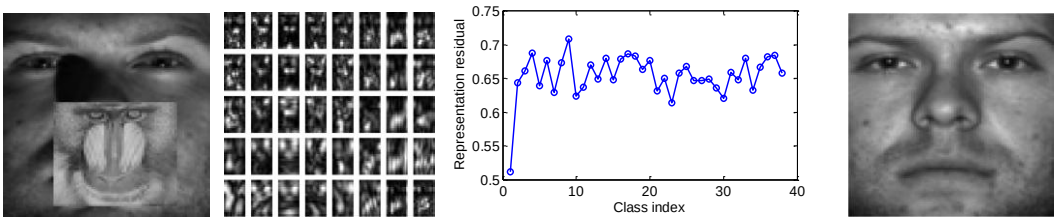
In this sub-section, we test the robustness of GRRC to face occlusions, including block occlusion and real disguise. FR with random block occlusion is performed on the Extended Yale B database [99, 206], while FR with real disguise is performed on the AR database [212].

1) FR with random block occlusion: As in [102], we chose Subsets 1 and 2 (717 images, normal-to-moderate lighting conditions) for training, and Subset 3 (453 images, more extreme lighting conditions) for testing. In accordance to the experiments in [102], the images were resized to 96×84, and the occlusion dictionary $\boldsymbol{A}_e$ in SRC is set to an

identity matrix.

With the above settings, in SRC the size of matrix $\boldsymbol{B}$ in Eq. (4-3) is 8064×8761. In the proposed GRRC, the dimension of augmented Gabor-feature vector is 8960 (40×16×14, $\rho \approx 40$). The GOD $\boldsymbol{\Gamma}$ is then computed using Algorithm in Table 4.2. In the experiment, we set the number of atoms in $\boldsymbol{\Gamma}$ to 200 (i.e., q=200, with compression ratio about 40:1), and hence the size of dictionary $\boldsymbol{B}_{\boldsymbol{\Gamma}}$ in Eq. (4-15) is 8960×917. Compared with the original SRC, the dictionary size of GRRC is reduced from 8761 to 917.

As in [102], we simulated various levels of contiguous occlusion, form 0% to 50%, by replacing a randomly located square block in each testing image with an unrelated image, whose size is determined by the occlusion percentage. The location of occlusion was randomly chosen for each testing image and is unknown to the computer. Fig. 4.8 illustrates the classification process by using an example. Fig. 4.8 (a) shows a testing image with 30% randomly located occlusion; Fig. 4.8 (b) shows the augmented Gabor features of the testing image. The residuals of GRRC_L_2 associated to all classes are plotted in Fig. 4.8(c), and a template image of the identified subject is shown in Fig. 4.8(d). The detailed recognition rates of GRRC, SRC, CRC and PCA+NN (used as the baseline) are listed in the Table 4.12. We see that GRRC can correctly classify all the testing images when the occlusion percentage is less than or equal to 30%. When the occlusion percentage becomes larger, the advantage of GRRC over SRC is getting higher. Especially, GRRC_L_1 can still have a recognition rate of 87.4% when half of image is occluded, while SRC and CRC only achieve a rate of 65.3% and 61.0 respectively. PCA+NN gets the worst results for it does not consider the occlusion. We could also see that good performance is still achieved when the representation coefficients on Gabor occlusion dictionary are regularized by l_2-norn in GRRC_L_2.



(a) (b) (c) (d)

Figure 4.8: An example of face recognition with block occlusion. (a) A 30% occluded test face image $\boldsymbol{y}$ from the first class of Extended Yale B. (b). Uniformly down-sampled Gabor features $\boldsymbol{\chi}(\boldsymbol{y})$ of the testing image. (c). Estimated residuals $r_i(\boldsymbol{y})$, i = 1, 2, …, 38. (d). One sample of the class to which the testing image is classified.

Table 4.12: The recognition rates (%) of different methods under different levels of block occlusion.

Occlusion ratio	0%	10%	20%	30%	40%	50%
SRC[102]	**100**	**100**	99.8	98.5	90.3	65.3
CRC	**100**	99.8	96.7	86.3	74.8	61.0
PCA+NN[102]	92.5	90.7	84.0	73.5	61.5	45.0
GRRC_L$_1$	**100**	**100**	**100**	**100**	96.5	**87.4**
GRRC_L$_2$	**100**	**100**	**100**	**100**	**97.1**	84.1

2) FR with real disguise: A subset from the AR database was used in this experiment. This subset consists of 1199 images from 100 subjects (14 samples each class except for a corrupted image w-027-14.bmp), 50 male and 50 female. 799 images (about 8 samples per subject) of non-occluded frontal views with various facial expressions were used for training, while the others for testing. The images are resized to 83×60. So in original SRC, the size of matrix $\boldsymbol{B}$ in Eq. (4-3) is 4980×5779. In the proposed GRRC, the dimension of Gabor-feature vectors is 5200 (40×13×10, ρ≈38), and 100 atoms (with compression ratio 50:1) are computed to form the GOD by Algorithm in Table 4.2. Thus the size of dictionary $\boldsymbol{B}_r$ in Eq. (4-15) is 5200×899, and the dictionary size is reduced from 5779 to 899 for GRRC.

We consider two separate testing sets of 200 images (1 sample each session and each subject, with neural expression). The first testing set contains images of the subjects wearing sunglasses, which occlude roughly 20% of the image. The second testing set is composed of images of the subjects wearing a scarf, which occlude roughly 40% of the images. The results by GRRC, SRC, CRC, PCA+NN and SVM are listed in Table 4.13

(where the results of SRC and PCA+NN are copied from the original paper [102]). We see that on faces occluded by sunglasses, GRRC achieves a recognition rate of 93.0%, over 5% higher than that of SRC, while for occlusion by scarves, the proposed GRRC_L_1 (GRRC_L_2) achieves a recognition rate of 79% (77.5%), about 20% higher than that of SRC. It is surprising that CRC gets 90.5% in the scarf case but with very low recognition accuracy in sunglass case. SVM gets bad performance for that it cannot learn the occlusion information from the training set without occlusion.

In [102], the authors also partitioned the image into blocks for face classification by assuming that the occlusion is continuous. Such an SRC scheme is denoted by SRC-p, with the CRC scheme denoted by CRC-p. Here, after partitioning the image into several blocks, we calculate the Gabor features of each block and then use GRRC to classify each block image. The final classification result is obtained by voting. We denote by GRRC-p the GRRC with partitioning. In experiments, as [102] we partitioned the images into eight (4×2) blocks of size 20×30. The Gabor-feature vector of each block is of dimension 800, and the number of atoms in the computed GOD $\boldsymbol{\Gamma}$ is set to 20. Thus the dictionary $\boldsymbol{B}$ in SRC is of size 600×1379, while the dictionary $\boldsymbol{B}_{\Gamma}$ in GRRC is of size 800×819. The recognition rates of SRC-p, CRC-p and GRRC-p are also listed in Table 4.13. We see that with partitioning, GRRC can lead to recognition rates of 100% on sunglasses and 99% on scarves, also better than SRC and CRC.

Table 4.13: Recognition rates (%) on the AR database with disguise occlusion ('-p': partitioned, '-sg': sunglasses, and '-sc': scarves).

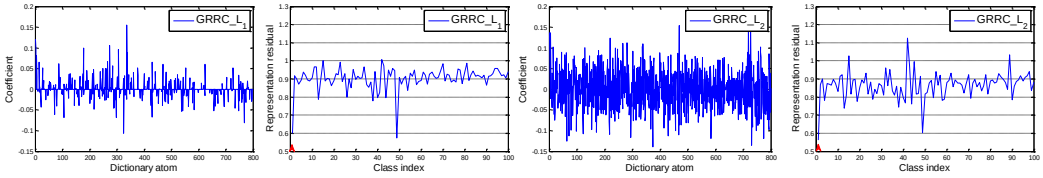
	Sunglass		Scarf	
SRC (SRC-p) [102]	87	(97.5)	59.5	(93.5)
CRC (CRC-p) [180]	68.5	(91.5)	**90.5**	(95)
PCA+NN [102]	70.0		12.0	
SVM	66.5		16.5	
GRRC_L_1 (GRRC-p_L_2)	**93.0**	**(100)**	**79.0**	(**99**)
GRRC_L_2 (GRRC-p_L_2)	**93.0**	**(100)**	77.5	(**99**)

3) Running time comparison: The recognition rates and running time of the proposed GRRC and SRC on a more challenging FR experiment with real disguise are compared here. A subset of 50 males and 50 females are selected from the AR database. For each subject, 7 samples with no occlusion from session 1 are used for training, with all the remaining samples with disguises for testing. These testing samples (including 3 sunglass samples in Session1, 3 sunglass samples in Session 2, 3 scarf samples in Session 1 and 3 scarf samples in Session 2 per subject) not only have disguises, but also have variations of time and illumination. The image size and the extraction of Gabor feature of GRRC remains the same as before. Here λ=0.005 for GRRC and the programming environment is Matlab version R2011a. The desktop used is of 1.86 GHz CPU and with 2.99G RAM. All the l_1-minimization problem is solved by using the fast solver: ALM [126, 202]. The recognition rates and running time of GRRC and SRC are listed in Table 4.14. The recognition rates of GRRC in all cases are much higher than SRC and CRC, especially with over 7% improvement on FR with sunglasses of session 1, and at least 43% in FR with scarf. It can also be seen that GRRC_L_1 is slightly better in FR with scarf, while GRRC_L_2 slightly better in FR with sunglasses. Fig. 4.9 plots the representation coefficients and residuals of a sample from class 1. As shown in Fig. 4.9(b), the sample is wrongly classified by GRRC_L_1 though the coefficients are sparse (see Fig. 4.9(a)). Although the representation coefficients of GRRC_L_2 are dense (Fig. 4.9(c)), the sample is correctly classified, as shown in Fig. 4.9(d).

Table 4.14: Recognition rates (%) and average running time (second) of GRRC and SRC on FR with disguise. (A-time: average time.)

	Sunglass-S1	Scarf-S1	Sunglass-S2	Scarf-S2	A-time	Speedup
SRC	83.3	48.7	49.0	29.0	12.278	--
CRC	78.0	52.3	44.7	29.3	0.084	146.2
GRRC_L_1	90.7	**95.3**	50.3	**87.3**	1.539	7.98
GRRC_L_2	**92.3**	95	**51.7**	84.3	0.331	37.09

The running time of SRC per testing sample is about 12 seconds, while GRRC_L_1 only needs about 1.5 seconds. However, this is still long for practical FR system. With l_2-norm regularization on the Gabor feature representation coefficients, the running time of GRRC_L_2 is only about 0.3 second, where 0.29 second is the running time of Gabor feature extraction. Although CRC is the fastest one, its recognition rate is also very low, similar to that of SRC. The speedup of GRRC_L_2 and GRRC_L_1 over SRC are 37.09 and 7.98 times, respectively.



(a) (b) (c) (d)

Figure 4.9: Representation coefficient and residual of a sample from class 1. (a) and (c) plot the coefficients of GRRC_L_1 and GRRC_L_2, respectively; (b) and (d) illustrate the representation residual associated to each class by GRRC_L_1 and GRRC_L_2, respectively.

It can be seen from the FR experiments with occlusion that GRRC could achieve much higher recognition accuracy than SRC and CRC. More importantly, with Gabor transformation the occlusion dictionary could be compressed, which reduces significantly the number of unknown parameters and the computational burden.

4.5 Discussion of Regularization on Coding Coefficients

In this section, we discuss the effect of feature dimension on the regularization (l_1-norm or l_2-norm) of coding coefficient. Fig. 4.10 plots the recognition rates of GRRC_L_1 and GRRC_L_2 versus different feature dimensionality with the same experiment setting on Mulit-PIE database in Section 4.4.3. The number of dictionary atoms is 3486 (14×249). From Fig. 10, we get that when the feature dimension is too low compared to the number

of dictionary atoms, GRRC_L_1 has better performance than GRRC_L_2. However, as the feature dimensionality increases, their recognition rates will become close. This phenomenon is consistent with the analysis of the l_1-norm and l_2-norm regularization on coding coefficients summarized in Chapter 3.

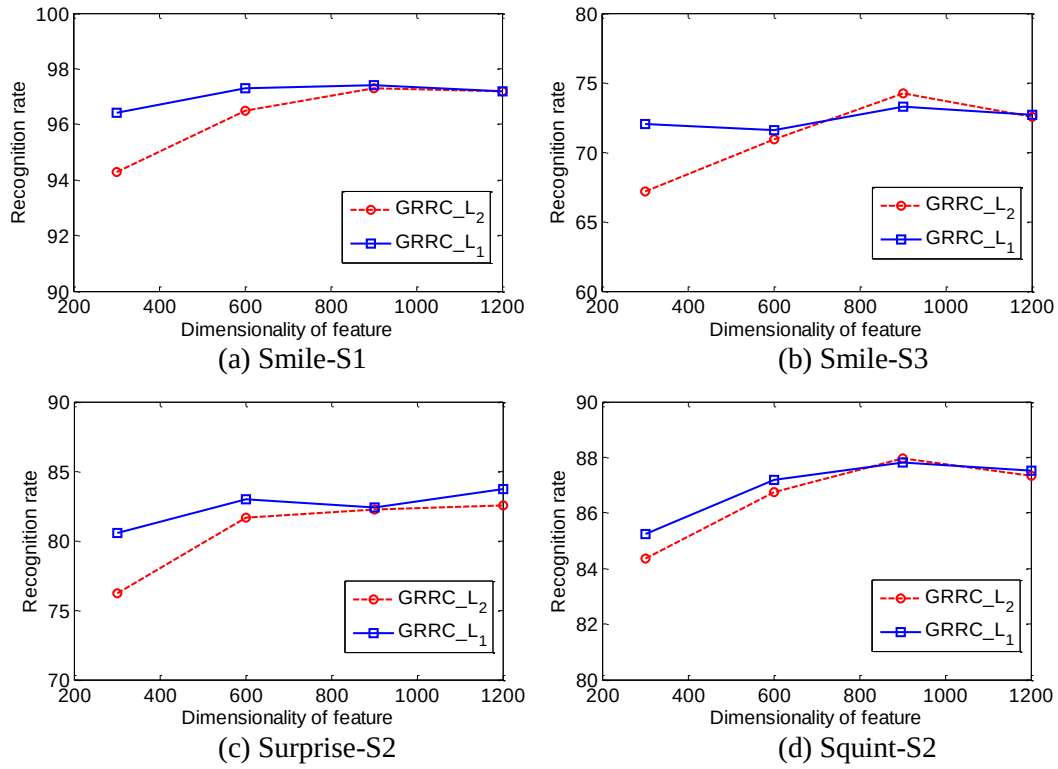


Figure 4.10: Recognition rates of GRRC_L_1 and GRRC_L_2 versus feature dimensionality in FR with expression variations. (a) FR with smile in session 1 for testing. (b) FR with smile in session 3 for testing. (c) FR with surprise in session 2 for testing. (d) FR with squint in session 2 for testing.

4.6 Summary

In this Chapter, we proposed a Gabor-feature based robust representation and classification (GRRC) scheme for face recognition, and proposed an associated Gabor occlusion dictionary (GOD) computing algorithm to handle the occluded face images. Apart from the improved face recognition rate, one important advantage of GRRC is its compact occlusion dictionary, which has much less atoms than that of the original SRC scheme. More importantly, we proposed to regularize the coding coefficients on the learnt GOD by

l_2-norm. This greatly reduces the computational cost of coding. We evaluated the proposed method on different conditions, including variations of illumination, expression and pose, as well as block occlusion and disguise occlusion. The experimental results clearly demonstrated that the proposed GRRC has much better performance than SRC, leading to much higher recognition rates while spending much less computational cost. This makes it much more practical to use than SRC in real world face recognition.

Chapter 5. Regularized Robust Coding for Face Recognition

5.1 Introduction

Face recognition (FR) to occlusion/corruption is a very challenging issue because of the variations of occlusion, such as disguise, continuous or pixel-wise occlusion, randomness of occlusion position and the intensity of occluded pixels. Several robust FR methods, such as LBP [90, 149], Eigenimages [230-231], probabilistic local approach [232], support vector machines based FR [233], and image gradient orientations for FR [86], have been proposed to deal with the occlusion in facial images. However, either only special kinds of occlusions can be handled (e.g., continuous occlusion [86, 149, 232] and occlusion in color face image [233]), or the performance is not satisfactory [230-231].

The recognition of a testing face image is usually accomplished by classifying the features extracted from this image. The most popular classifiers for FR may be the nearest neighbor (NN) classifier and its variants, e.g., nearest feature line (NFL) [96], nearest feature plane (NFP) [97], and nearest subspace (NS) [97-101], all of which aim to find a suitable representation of the testing face image, and classify it by checking which class can give a better representation than other classes. Though NFL, NSP, and NS may achieve better performance than NN, they are not robust to face occlusion with holistic face features. Therefore how to formulate the representation model for classification tasks such as FR is still a challenging problem.

In recent years, sparse representation (or sparse coding) has been attracting a lot of attention due to its great success in image processing [131-135], and it has also been used for FR [102, 146, 181, 234] and texture classification [144, 162]. In general, the sparse coding problem can be formulated as

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \text{ s.t. } \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \le \varepsilon \tag{5-1}$$

where $\boldsymbol{y}$ is the given signal, $\boldsymbol{D}$ is the dictionary of coding atoms, $\boldsymbol{\alpha}$ is the coding vector of $\boldsymbol{y}$ over $\boldsymbol{D}$, and $\varepsilon > 0$ is a constant. Recently, Wright *et al.* [102] applied sparse coding to FR and proposed the sparse representation based classification (SRC) scheme. By coding a testing image $\boldsymbol{y}$ as a sparse linear combination of all the training samples via Eq. (5-1), SRC classifies $\boldsymbol{y}$ by evaluating which class could result in the minimal reconstruction error of it. One interesting feature of SRC is its processing of face occlusion and corruption. More specifically, it introduces an identity matrix $\boldsymbol{I}$ as a dictionary to code the outlier pixels (e.g., corrupted or occluded pixels):

$$\min_{\boldsymbol{\alpha},\boldsymbol{e}} \|[\boldsymbol{\alpha};\boldsymbol{e}]\|_1 \text{ s.t. } \boldsymbol{y}=[\boldsymbol{D},\boldsymbol{I}]\cdot[\boldsymbol{\alpha};\boldsymbol{e}] \tag{5-2}$$

By solving Eq. (5-2), SRC shows good robustness to face occlusions such as block occlusion, pixel corruption and disguise. It is not difficult to see that Eq. (5-2) is basically equivalent to $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_1 < \varepsilon$. That is, it uses l_1-norm to model the coding residual $\boldsymbol{y}$-$\boldsymbol{D}\boldsymbol{\alpha}$ to gain certain robustness to outliers.

Although the sparse coding model in Eq. (5-1) has made a great success in image restoration [131-135] and led to interesting results in FR [102, 146, 181, 234], there are two issues to be considered more carefully when applying it to pattern classification tasks such as FR. One is that whether the l_1-sparsity constraint $\|\cdot\|_1$ is indispensable to regularize the solution, since the l_1-minimization needs much computational cost. The other is that whether the term $\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \le \varepsilon$ is effective enough to characterize the signal fidelity, especially when the observation $\boldsymbol{y}$ is noisy and/or has many outliers. For the first issue, on one side reweighted l_1 or l_2 minimization was proposed to speed up the sparse coding process [235-236]; one the other side some works [180, 237-238] have questioned the use of sparse coding for image classification. Particularly, as shown in Chapter 3, it is not necessary to impose the l_1-sparsity constraint on the coding vector $\boldsymbol{\alpha}$, while the l_2-norm regularization on $\boldsymbol{\alpha}$ performs equally well. For the second issue, to the best of our

knowledge, few works have been reported in the scheme of sparse representation except for the l_1-norm fidelity (i.e., $\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_1 \le \varepsilon$) in [102, 234], and the correntropy based Gaussian-kernel fidelity in [239-240]. The fidelity term has a very high impact on the final coding result. From the viewpoint of *maximum a posterior* (MAP) estimation, defining the fidelity term with l_2- or l_1-norm actually assumes that the coding residual $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}$ follows Gaussian or Laplacian distribution. In practice, however, such an assumption may not hold well, especially when occlusions, corruptions and expression variations occur in the testing face images. Although Gaussian kernel based fidelity term utilized in [239-240] is claimed to be robust to non-Gaussian noise [241], it may not work well in FR with occlusion due to the complex variation of occlusion. For example, the scarf disguise occlusion needs to be manually removed in [240].

To increase the robustness of FR to occlusion, pixel corruption, disguises and big expression variations, etc., we propose a regularized robust coding (RRC) model in this Chapter. We assume that the coding residual $\boldsymbol{e}$ and the coding vector $\boldsymbol{\alpha}$ are respectively independent and identically distributed, and then robustly regress the given signal based on the MAP principle. In implementation, the RRC minimization problem is transformed into an iteratively reweighted regularized robust coding (IR^3C) problem with a reasonably designed weight function for robust FR. Our extensive experiments in benchmark face databases show that RRC achieves much better performance than existing sparse representation based FR methods, especially when there are complicated variations, such as face occlusions, corruptions and expression changes, etc.

5.2 Regularized Robust Coding (RRC)

5.2.1 The modeling of RRC

The conventional sparse coding model in Eq. (5-1) is equivalent to the so-called LASSO

problem [117]:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \text{ s.t. } \|\boldsymbol{\alpha}\|_1 \le \sigma \tag{5-3}$$

where $\sigma > 0$ is a constant, $\boldsymbol{y}=[y_1;y_2;\ldots;y_n] \in \Re^n$ is the signal to be coded, $\boldsymbol{D}=[\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_m] \in \Re^{n\times m}$ is the dictionary with column vector $\boldsymbol{d}_j$ being its j^{th} atom, and $\boldsymbol{\alpha} \in \Re^m$ is the vector of coding coefficients. In the problem of face recognition (FR), the atom $\boldsymbol{d}_j$ can be simply set as the training face sample (or its dimensionality reduced feature) and hence the dictionary $\boldsymbol{D}$ can be the whole training dataset.

If we have the prior that the coding residual $\boldsymbol{e} = \boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}$ follows Gaussian distribution, the solution to Eq. (5-3) will be the maximum likelihood estimation (MLE) solution. If $\boldsymbol{e}$ follows Laplacian distribution, the l_1-sparsity constrained MLE solution will be

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_1 \text{ s.t. } \|\boldsymbol{\alpha}\|_1 \le \sigma \tag{5-4}$$

The above Eq. (5-4) is essentially another expression of Eq. (5-2) because they have the same Lagrangian formulation: $\min_{\boldsymbol{\alpha}}\{\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}\|_1+\lambda\|\boldsymbol{\alpha}\|_1\}$ [202].

In practice, however, the Gaussian or Laplacian priors on $\boldsymbol{e}$ may be invalid, especially when the face image $\boldsymbol{y}$ is occluded, corrupted, etc. Let's use examples to illustrate the fitted distributions of residual $\boldsymbol{e}$ by different models. Fig. 5.1(a) shows a clean face image, denoted by $\boldsymbol{y}_o$, while Fig. 5.1(b) and Fig. 5.1(c) show the occluded and corrupted testing images $\boldsymbol{y}$, respectively. The residual is computed as $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{D}\hat{\boldsymbol{\alpha}}$, while to make the coding vector more accurate we use the clean image to calculate it via Eq. (5-3): $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y}_o-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \text{ s.t.} \|\boldsymbol{\alpha}\|_1 \le \sigma$. The empirical and fitted distributions of $\boldsymbol{e}$ by using Gaussian, Laplacian and the distribution model (refer to Eq. (5-15)) associated with the proposed method are plotted in Fig. 5.1(d). Fig. 5.1(e) shows the distributions in log domain for better observation of the tails. It can be seen that the empirical distribution of $\boldsymbol{e}$ has a strong peak at zero but a long tail, which is mostly caused by the occluded and corrupted pixels. For robust FR, a good fitting of the tail is much more important than the fitting of the peak, which is produced by the small trivial coding errors. It can be seen

from Fig. 5.1(e) that the proposed model can well fit the heavy tail of the empirical distribution, much better than the Gaussian and Laplacian models. Meanwhile, Laplacian works better than Gaussian in fitting the heavy tail, which explains why the sparse coding model in Eq. (5-4) (or Eq. (5-2)) works better than the model in Eq. (5-1) (or Eq. (5-3)) in handling face occlusion and corruption.

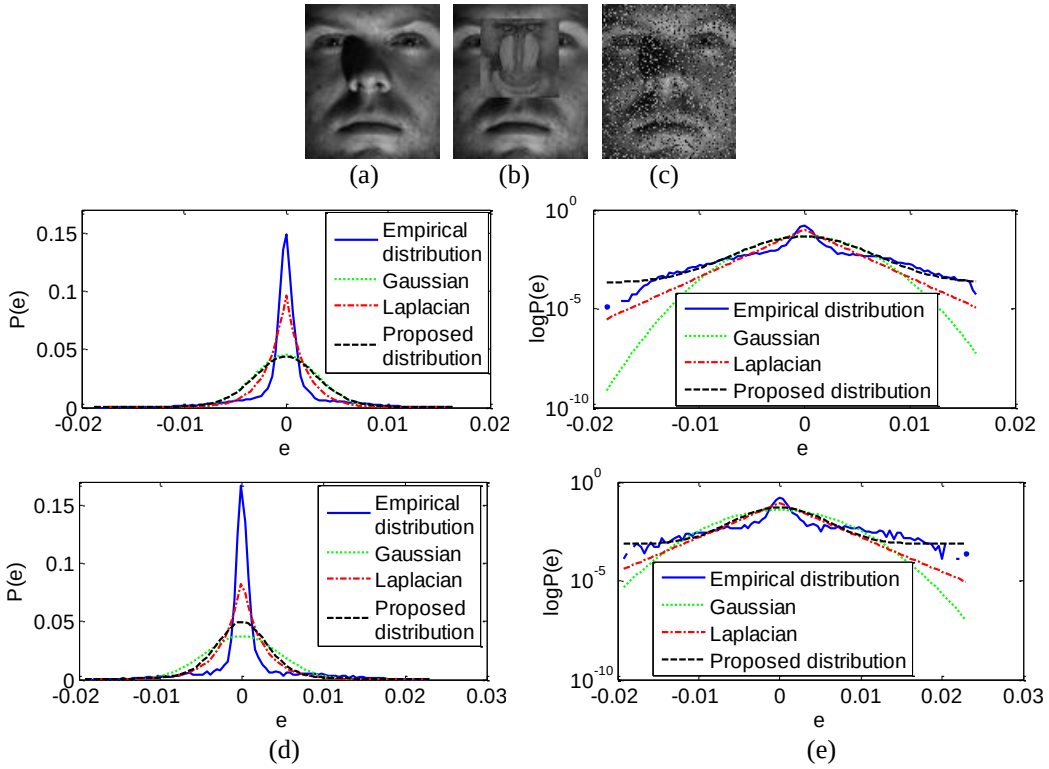


Figure 5.1: The empirical distribution of coding residuals and the fitted distributions by different models. (a) Clean face image; (b) and (c) are occluded and corrupted testing face images; (d) and (e) show the distributions (top row: occluded image; bottom row: corrupted image) of coding residuals in linear and log domains, respectively.

Inspired by the robust regression theory [242-244], in our work [195] we proposed an MLE solution for robust face image representation. Rewrite $\boldsymbol{D}$ as $\boldsymbol{D} = [\boldsymbol{r}_1; \boldsymbol{r}_2; \ldots; \boldsymbol{r}_n]$, where $\boldsymbol{r}_i$ is the i^{th} row of $\boldsymbol{D}$, and let $\boldsymbol{e} = \boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha} = [e_1; e_2; \ldots; e_n]$, where $e_i = y_i-\boldsymbol{r}_i\boldsymbol{\alpha}$, i=1,2,…,n. Assume that $e_1, e_2, \ldots, e_n$ are independent and identically distributed (i.i.d.) and the probability density function (PDF) of e_i is $f_{\theta}(e_i)$, where $\boldsymbol{\theta}$ denotes the unknown parameter set that characterizes the distribution, the so-called robust sparse coding (RSC) [195] was

formulated as the following l_1-sparsity constrained MLE problem (let $\rho_\theta(e) = -\ln f_\theta(e)$)

$$\min_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \rho_\theta\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}\right) \text{ s.t. } \|\boldsymbol{\alpha}\|_1 \le \sigma \tag{5-5}$$

Like SRC, the above RSC model assumes that the coding coefficients are sparse and uses l_1-norm to characterize the sparsity. However, the l_1-sparsity constraint makes the complexity of RSC high, and recently it has been indicated in [180] that the l_1-sparsity constraint on $\boldsymbol{\alpha}$ is not the key for the success of SRC [102]. Therefore we then propose a more general model, namely regularized robust coding (RRC). The RRC can be much more efficient than RSC, while RSC is one specific instantiation of the RRC model.

Let's consider the face representation problem from a viewpoint of Bayesian estimation, more specifically, the *maximum a posterior* (MAP) estimation. By coding the testing image $\boldsymbol{y}$ over a given dictionary $\boldsymbol{D}$, the MAP estimation of the coding vector $\boldsymbol{\alpha}$ is $\hat{\boldsymbol{\alpha}} = \arg\max_{\boldsymbol{\alpha}} \ln P\left(\boldsymbol{\alpha} \mid \boldsymbol{y}\right)$. Using the Bayesian formula, we have

$$\hat{\boldsymbol{\alpha}} = \arg\max_{\boldsymbol{\alpha}} \left\{\ln P\left(\boldsymbol{y} \mid \boldsymbol{\alpha}\right) + \ln P\left(\boldsymbol{\alpha}\right)\right\} \tag{5-6}$$

Assuming that the elements e_i of coding residual $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{D}\boldsymbol{\alpha}$ = [e_1; e_2; …; e_n] are i.i.d. with PDF $f_\theta(e_i)$, we have $P\left(\boldsymbol{y} \mid \boldsymbol{\alpha}\right) = \prod_{i=1}^{n} f_\theta\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}\right)$. Meanwhile, assume that the elements α_j, j=1,2,…,m, of the coding vector $\boldsymbol{\alpha}$=[α_1; α_2; …; α_m] are i.i.d. with PDF $f_o(\alpha_j)$, there is $P\left(\boldsymbol{\alpha}\right) = \prod_{j=1}^{m} f_o\left(\alpha_j\right)$. The MAP estimation of $\boldsymbol{\alpha}$ in Eq. (5-6) is

$$\hat{\boldsymbol{\alpha}} = \arg\max_{\boldsymbol{\alpha}} \left\{\ln \prod_{i=1}^{n} f_\theta\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}\right) + \ln \prod_{j=1}^{m} f_o\left(\alpha_j\right)\right\} \tag{5-7}$$

Letting $\rho_\theta(e) = -\ln f_\theta(e)$ and $\rho_o(\alpha) = -\ln f_o(\alpha)$, Eq. (5-7) is converted into

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{\sum_{i=1}^{n} \rho_\theta\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}\right) + \sum_{j=1}^{m} \rho_o\left(\alpha_j\right)\right\} \tag{5-8}$$

We call the above model regularized robust coding (RRC) because the fidelity term $\sum_{i=1}^{n} \rho_\theta\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}\right)$ will be very robust to outliers, while $\sum_{j=1}^{m} \rho_o\left(\alpha_j\right)$ is the regularization term depending on the prior probability $P(\boldsymbol{\alpha})$.

It can be seen that $\sum_{j=1}^{m} \rho_o\left(\alpha_j\right)$ becomes the l_1-norm sparse constraint when α_j is

Laplacian distributed, i.e., $P(\boldsymbol{\alpha}) = \prod_{j=1}^{m} \exp\left(-\left\|\alpha_j\right\|_1 / \sigma_\alpha\right) / 2\sigma_\alpha$. For the problem of classification, it is desired that only the representation coefficients associated with the dictionary atoms from the target class could have big absolute values. As we do not know beforehand which class the testing image belongs to, a reasonable prior can be that only a small percent of representation coefficients have significant values. Therefore, we assume that the representation coefficient α_j follows generalized Gaussian distribution (GGD). There is

$$f_o\left(\alpha_j\right) = \beta \exp\left\{-\left(\left|\alpha_j\right| / \sigma_\alpha\right)^\beta\right\} / \left(2\sigma_\alpha \Gamma\left(1/\beta\right)\right) \tag{5-9}$$

where Γ denotes the gamma function.

For the representation residual, it is difficult to predefine the distribution due to the diversity of image variations. In general, we assume that the unknown PDF $f_\theta(e)$ are symmetric, differentiable, and monotonic w.r.t. $|e|$, respectively. So $\rho_\theta(e)$ has the following properties: (1) $\rho_\theta(0)$ is the global minimal of $\rho_\theta(x)$; (2) symmetry: $\rho_\theta(x)= \rho_\theta(-x)$; (3) monotonicity: $\rho_\theta(x_1)> \rho_\theta(x_2)$ if $|x_1|>|x_2|$. Without loss of generality, we let $\rho_\theta(0)=0$.

Two key issues in solving the RRC model are how to determine the distributions ρ_θ (or f_θ), and how to minimize the energy functional. Simply taking f_θ as Gaussian or Laplacian and taking f_o as Laplacian, the RRC model will degenerate to the conventional sparse coding problem in Eq. (5-3) or Eq. (5-4). However, as we showed in Fig. 5.1, such preset distributions for f_θ have much bias and are not robust enough to outliers, and the Laplacian setting of f_o makes the minimization inefficient. In this Chapter, we allow f_θ to have a more flexible shape, which is adaptive to the input testing image $\boldsymbol{y}$ so that the system is more robust to outliers. To this end, we transform the minimization of Eq. (5-8) into an iteratively reweighted regularized coding problem in order to obtain the approximated MAP solution of RRC effectively and efficiently.

5.2.2 RRC via iteratively reweighting

Let $F_{\boldsymbol{\theta}}(\boldsymbol{e}) = \sum_{i=1}^{n} \rho_{\boldsymbol{\theta}}(e_i)$. The Taylor expansion of $F_{\boldsymbol{\theta}}(\boldsymbol{e})$ in the neighborhood of $\boldsymbol{e}_0$ is:

$$\tilde{F}_{\boldsymbol{\theta}}(\boldsymbol{e}) = F_{\boldsymbol{\theta}}(\boldsymbol{e}_0) + (\boldsymbol{e} - \boldsymbol{e}_0)^T F'_{\boldsymbol{\theta}}(\boldsymbol{e}_0) + R_1(\boldsymbol{e}) \tag{5-10}$$

where $R_1(\boldsymbol{e})$ is the high order residual, and $F'_{\boldsymbol{\theta}}(\boldsymbol{e})$ is the derivative of $F_{\boldsymbol{\theta}}(\boldsymbol{e})$. Denote by $\rho'_{\boldsymbol{\theta}}$ the derivative of $\rho_{\boldsymbol{\theta}}$, and there is $F'_{\boldsymbol{\theta}}(\boldsymbol{e}_0) = \left[\rho'_{\boldsymbol{\theta}}(e_{0,1}); \rho'_{\boldsymbol{\theta}}(e_{0,2}); \cdots; \rho'_{\boldsymbol{\theta}}(e_{0,n})\right]$, where $e_{0,i}$ is the i^{th} element of $\boldsymbol{e}_0$. To make $F'_{\boldsymbol{\theta}}(\boldsymbol{e})$ strictly convex for easier minimization, we approximate the residual term as $R_1(\boldsymbol{e}) \approx 0.5(\boldsymbol{e}-\boldsymbol{e}_0)^T \boldsymbol{W}(\boldsymbol{e}-\boldsymbol{e}_0)$, where $\boldsymbol{W}$ is a diagonal matrix for that the elements in $\boldsymbol{e}$ are independent and there is no cross term of e_i and e_j, $i \neq j$, in $F_{\boldsymbol{\theta}}(\boldsymbol{e})$.

Since $F_{\boldsymbol{\theta}}(\boldsymbol{e})$ reaches its minimal value (i.e., 0) at $\boldsymbol{e}=\boldsymbol{0}$, we also require that its approximation $\tilde{F}_{\boldsymbol{\theta}}(\boldsymbol{e})$ reaches the minimum at $\boldsymbol{e}=\boldsymbol{0}$. Letting $\tilde{F}'_{\boldsymbol{\theta}}(\boldsymbol{0})=0$, we have the diagonal elements of $\boldsymbol{W}$ as

$$\boldsymbol{W}_{i,i} = \omega_{\boldsymbol{\theta}}(e_{0,i}) = \rho'_{\boldsymbol{\theta}}(e_{0,i}) / e_{0,i} \tag{5-11}$$

According to the properties of $\rho_{\boldsymbol{\theta}}$, we know that $\rho'_{\boldsymbol{\theta}}(e_i)$ will have the same sign as e_i. So $\boldsymbol{W}_{i,i}$ is a non-negative scalar. Then $\tilde{F}_{\boldsymbol{\theta}}(\boldsymbol{e})$ can be written as

$$\tilde{F}_{\boldsymbol{\theta}}(\boldsymbol{e}) = \frac{1}{2}\left\|\boldsymbol{W}^{1/2}\boldsymbol{e}\right\|_2^2 + b_{\boldsymbol{e}_0} \tag{5-12}$$

where $b_{\boldsymbol{e}_0} = \sum_{i=1}^{n}\left[\rho_{\boldsymbol{\theta}}(e_{0,i}) - \rho'_{\boldsymbol{\theta}}(e_{0,i})e_{0,i}/2\right]$ is a scalar constant determined by $\boldsymbol{e}_0$.

Without considering the constant $b_{\boldsymbol{e}_0}$, the RRC model in Eq. (5-8) could be approximated as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2}\left\|\boldsymbol{W}^{1/2}(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha})\right\|_2^2 + \sum_{j=1}^{m} \rho_{\boldsymbol{o}}(\alpha_j) \right\} \tag{5-13}$$

Certainly, Eq. (5-13) is a local approximation of Eq. (5-8) but it makes the minimization of RRC feasible via iteratively reweighted l_2-regularized coding, in which $\boldsymbol{W}$ is updated

via Eq. (5-11). Now, the minimization of RRC is turned to how to calculate the diagonal weight matrix $\boldsymbol{W}$.

5.2.3 The weights $\boldsymbol{W}$

The element $\boldsymbol{W}_{i,i}$, i.e., $\omega_{\theta}(e_i)$, is the weight assigned to pixel i of testing image $\boldsymbol{y}$. Intuitively, in FR the outlier pixels (e.g., occluded or corrupted pixels) should have small weights to reduce their effect on coding $\boldsymbol{y}$ over $\boldsymbol{D}$. Since the dictionary $\boldsymbol{D}$, composed of non-occluded/non-corrupted training face images, could well represent the facial parts, the outlier pixels will have rather big coding residuals. Thus, the pixel which has a big residual e_i should have a small weight. Such a principle can be observed from Eq. (5-11), where $\omega_{\theta}(e_i)$ is inversely proportional to e_i and modulated by $\rho'_{\theta}(e_i)$. Refer to Eq. (5-11), since ρ_{θ} is differentiable, symmetric, monotonic and has its minimum at origin, we can assume that $\omega_{\theta}(e_i)$ is continuous and symmetric, while being inversely proportional to e_i but bounded (to increase stability). Without loss of generality, we let $\omega_{\theta}(e_i)\in[0, 1]$. With these considerations, one good choice of $\omega_{\theta}(e_i)$ is the widely used logistic function [245]:

$$\omega_{\theta}(e_i)=\exp\left(-\mu e_i^2+\mu\delta\right)\Big/\left(1+\exp\left(-\mu e_i^2+\mu\delta\right)\right) \tag{5-14}$$

where μ and δ are positive scalars. Parameter μ controls the decreasing rate from 1 to 0, and δ controls the location of demarcation point. Here the value of $\mu\delta$ should be big enough to make $\omega_{\theta}(0)$ close to 1 (usually we set $\mu\delta\geqslant 8$). With Eq. (5-14), Eq. (5-11) and $\rho_{\theta}(0)=0$, we could get

$$\rho_{\theta}(e_i)=-\frac{1}{2\mu}\left(\ln\left(1+\exp\left(-\mu e_i^2+\mu\delta\right)\right)-\ln\left(1+\exp\mu\delta\right)\right) \tag{5-15}$$

We can see that the above ρ_{θ} satisfies all the assumptions and properties discussed in Section 5.2.1.

The PDF f_{θ} associated with ρ_{θ} in Eq.(5-15) is more flexible than the Gaussian and Laplacian functions to model the residual $\boldsymbol{e}$. It can have a longer tail to address the

residuals yielded by outlier pixels such as corruptions and occlusions (refer to Fig. 5.1 for examples), and hence the coding vector $\boldsymbol{\alpha}$ will be robust to the outliers in $\boldsymbol{y}$. $\omega_{\theta}(e_i)$ could also be set as other functions. However, as indicated by [246], the proposed logistic weight function is the binary classifier derived via MAP estimation, which is suitable to distinguish inliers and outliers. When $\omega_{\theta}(e_i)$ is set as a constant such as $\omega_{\theta}(e_i)$=2, it corresponds to the l_2-norm fidelity in Eq. (5-3); when set as $\omega_{\theta}(e_i)=1/|e_i|$, it corresponds to the l_1-norm fidelity in Eq. (5-4); when set as a Gaussian function $\omega_{\theta}(e_i)=\exp(-e_i^2/2\sigma^2)$, it corresponds to the Gaussian kernel fidelity in [239-240]. However, all these functions are not as robust as Eq. (5-14) to outliers, as illustrated in Fig. 5.2. From Fig. 5.2, one can see that the l_2-norm fidelity treats all pixels equally, no matter it is outlier or not; the l_1-norm fidelity assigns higher weights to pixels with smaller residuals; however, the weight can be infinity when the residual approaches to zero, making the coding unstable. Both our proposed weight function and the weight function of the Gaussian fidelity used in [239-240] are bounded in [0, 1], and they have an intersection point with weight value as 0.5. However, the proposed weight function prefers to assign larger weights to inliers and smaller weights to outliers; that is, it has higher capability to classify inliers and outliers.

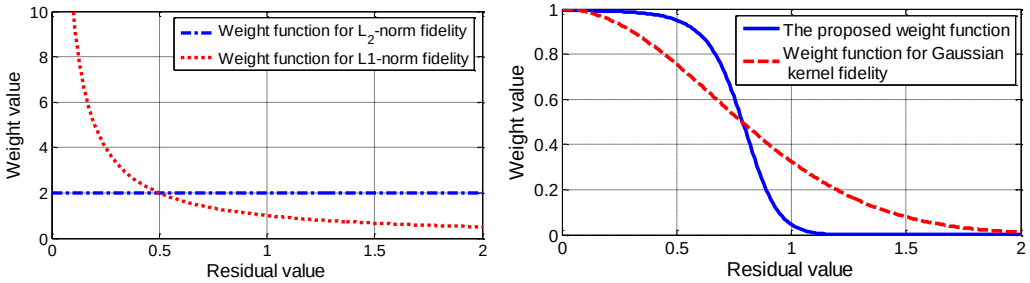


Figure 5.2: Weight functions for different signal fidelity terms, including (a) l_2 and l_1-norm fidelity terms in SRC [102] and (b) the Gaussian kernel fidelity term [239-240], as well as the proposed RRC fidelity term.

The sparse coding models in Eqs. (5-3) and (5-4) are instantiations of the RRC model with Eq.(5-14) and β=1 in Eq.(5-9). The model in Eq. (5-3) is the case by letting $\omega_\theta(e_i)$=2. The model in Eq. (5-4) is the case by letting $\omega_\theta(e_i)$=1/|e_i|. Compared with the models in Eqs. (5-3) and (5-4), the proposed RRC model (Eq. (5-8) or Eq. (5-13)) is much more robust to outliers (usually the pixels with big residuals) because it will adaptively assign small weights to them. Although the model in Eq. (5-4) also assigns small weights to outliers, its weight function $\omega_\theta(e_i)$=1/|e_i| is not bounded (i.e., the weights assigned to very small residuals can have very big values and dramatic changing ratios), making it less effective to distinguish between inliers and outliers.

5.2.4 Two important cases of RRC

The minimization of RRC model in Eq. (5-13) can be accomplished iteratively, while in each iteration $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ are updated alternatively. By fixing the weight matrix $\boldsymbol{W}$, the RRC with GGD prior on representation (Eq. (5-13)) and $\rho_\sigma(\alpha)$ =-ln$f_\sigma(\alpha)$ could be written as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \left\| \boldsymbol{W}^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha} \right) \right\|_2^2 + \sum_{j=1}^{m} \left(\lambda \left| \alpha_j \right|^\beta + b_{\alpha_0} \right) \right\} \tag{5-16}$$

where $\rho_\sigma\left(\alpha_j\right) = \lambda\left|\alpha_j\right|^\beta + b_{\alpha_0}$, $\lambda = \left(1/\sigma_\alpha\right)^\beta$ and $b_{\alpha_0} = \ln\left(2\sigma_\alpha \Gamma\left(1/\beta\right)/\beta\right)$ is a constant. Similar to the processing of $F_\theta(\boldsymbol{e}) = \sum_{i=1}^{n} \rho_\theta\left(e_i\right)$ in Section 5.2.2, $\sum_{j=1}^{m} \left|\alpha_j\right|^\beta$ could also be approximated by the Taylor expansion. Then Eq. (5-16) changes to

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \left\| \boldsymbol{W}^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha} \right) \right\|_2^2 + \sum_{j=1}^{m} \boldsymbol{V}_{j,j} \alpha_j^2 \right\} \tag{5-17}$$

where $\boldsymbol{W}$ is a diagonal matrix with $\boldsymbol{V}_{j,j} = \rho_\sigma'\left(\alpha_j\right)/\alpha_j$.

The value of β determines the types of regularization. If $0<\beta\leqslant 1$, then sparse regularization is applied; otherwise, non-sparse regularization is imposed on the representation coefficients. In particular, the proposed RRC model has two important cases with two specific values of β.

When β=2, GGD degenerates to the Gaussian distribution, and the RRC model becomes

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \left\| \boldsymbol{W}^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha} \right) \right\|_2^2 + \lambda \left\| \boldsymbol{\alpha} \right\|_2^2 \right\} \tag{5-18}$$

In this case the RRC model is essentially an l_2-regularized robust coding model. It can be easily derived that when $\boldsymbol{W}$ is given, the solution to Eq. (5-18) is $\hat{\boldsymbol{\alpha}} = \left(\boldsymbol{D}^T \boldsymbol{W} \boldsymbol{D} + \lambda \boldsymbol{I} \right)^{-1} \boldsymbol{D}^T \boldsymbol{W} \boldsymbol{y}$.

When β=1, GGD degenerates to the Laplacian distribution, and the RRC model becomes

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \left\| \boldsymbol{W}^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha} \right) \right\|_2^2 + \lambda \left\| \boldsymbol{\alpha} \right\|_1 \right\} \tag{5-19}$$

In this case the RRC model is essentially the RSC model in [195], where the sparse coding methods such as l_1_*ls* [204] is used to solve Eq. (19) when $\boldsymbol{W}$ is given. In this Chapter, we solve Eq. (5-19) via Eq. (5-17) by the iteratively re-weighting technique [235]. Let $\boldsymbol{V}_{j,j}^{(0)} = 1$, and then in the ($k$+1)$^{\text{th}}$ iteration the diagonal matrix $\boldsymbol{V}$ is set as $\boldsymbol{V}_{j,j}^{(k+1)} = \lambda \left| (\alpha_j^{(k)})^2 + \varepsilon^2 \right|^{-1/2}$, and then $\hat{\boldsymbol{\alpha}}^{(k+1)} = \left(\boldsymbol{D}^T \boldsymbol{W} \boldsymbol{D} + \boldsymbol{V}^{(k+1)} \right)^{-1} \boldsymbol{D}^T \boldsymbol{W} \boldsymbol{y}$. Here ε is a scalar defined in [235].

5.3 Algorithm of RRC

5.3.1 Iteratively reweighted regularized robust coding (IR^{3}C) algorithm

As discussed in Section 5.2, the minimization of RRC is an iterative process, and the weights $\boldsymbol{W}$ and $\boldsymbol{V}$ are updated alternatively in order for the desired coding vector $\boldsymbol{\alpha}$. Although we can only have a locally optimal solution to the RRC model, fortunately in FR we are able to have a very reasonable initialization to achieve good performance. In this section we propose an iteratively reweighted regularized robust coding (IR^{3}C)

algorithm to minimize the RRC model.

Table 5.1: Algorithm of Iteratively Reweighted Regularized Robust Coding.

Iteratively Reweighted Regularized Robust Coding (IR^{3}C)

Input: Normalized testing image $\boldsymbol{y}$ with unit l_2-norm; dictionary $\boldsymbol{D}$ (each column of $\boldsymbol{D}$ has unit l_2-norm); $\boldsymbol{\alpha}^{(1)}$.

Output: $\boldsymbol{\alpha}$

Start from t=1:

1. Compute residual $\boldsymbol{e}^{(t)} = \boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha}^{(t)}$.
2. Estimate weights as

$$\omega_{\boldsymbol{\theta}}\left(e_i^{(t)}\right) = 1 \Big/ 1 + \exp\left(\mu\left(e_i^{(t)}\right)^2 - \mu\delta\right),$$

   where μ and δ could be estimated in each iteration (please refer to Section 5.4.1 for the settings of them).
3. Weighted regularized robust coding:

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \left\| \left(\boldsymbol{W}^{(t)}\right)^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha}\right) \right\|_2^2 + \sum_{j=1}^{m} \rho_o\left(\alpha_j\right) \right\} \tag{5-21}$$

   where $\boldsymbol{W}^{(t)}$ is the estimated diagonal weight matrix with $\boldsymbol{W}_{i,i}^{(t)} = \omega_{\boldsymbol{\theta}}\left(e_i^{(t)}\right)$, $\rho_o\left(\alpha_j\right) = \lambda\left|\alpha_j\right|^{\beta} + b_{\alpha_0}$ and $\beta = 2$ or 1.
4. Update the sparse coding coefficients:
   If t=1, $\boldsymbol{\alpha}^{(t)} = \boldsymbol{\alpha}^*$;
   If t>1, $\boldsymbol{\alpha}^{(t)} = \boldsymbol{\alpha}^{(t-1)} + \upsilon^{(t)}\left(\boldsymbol{\alpha}^* - \boldsymbol{\alpha}^{(t-1)}\right)$;
   where $0 < \upsilon^{(t)} \le 1$ is a suitable step size that makes $\sum_{i=1}^{n} \rho_{\boldsymbol{\theta}}\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}^{(t)}\right) + \sum_{j=1}^{m} \rho_o\left(\alpha_j^{(t)}\right) < \sum_{i=1}^{n} \rho_{\boldsymbol{\theta}}\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}^{(t-1)}\right) + \sum_{j=1}^{m} \rho_o\left(\alpha_j^{(t-1)}\right)$. $\upsilon^{(t)}$ can be searched from 1 to 0 by the standard line-search process [247].
5. Compute the reconstructed testing sample:

$$\boldsymbol{y}_{rec}^{(t)} = \boldsymbol{D}\boldsymbol{\alpha}^{(t)},$$

   and let t=t+1.
6. Go back to step 1 until the condition of convergence (refer to Section 5.3.2) is met, or the maximal number of iterations is reached.

When a testing face image $\boldsymbol{y}$ comes, in order to initialize $\boldsymbol{W}$, we should firstly initialize the coding residual $\boldsymbol{e}$ of $\boldsymbol{y}$. We initialize $\boldsymbol{e}$ as $\boldsymbol{e}$=$\boldsymbol{y}$–$\boldsymbol{D}\boldsymbol{\alpha}^{(1)}$, where $\boldsymbol{\alpha}^{(1)}$ is an initial coding

vector. Because we do not know which class the testing face image $\boldsymbol{y}$ belongs to, a reasonable $\boldsymbol{\alpha}^{(1)}$ can be set as

$$\boldsymbol{\alpha}^{(1)} = \left[\tfrac{1}{m}; \tfrac{1}{m}; ...; \tfrac{1}{m}\right] \tag{5-20}$$

That is, $\boldsymbol{D}\boldsymbol{\alpha}^{(1)}$ is the mean image of all training samples. With the initialized coding vector $\boldsymbol{\alpha}^{(1)}$, the proposed IR^{3}C algorithm is summarized in Table 5.1.

When IR^{3}C converges, we use the same classification strategy as in SRC [18] to classify the face image $\boldsymbol{y}$:

$$\text{identity}(\boldsymbol{y}) = \arg\min_c \{\ell_c\} \tag{5-22}$$

where $\ell_c = \left\|\boldsymbol{W}_{final}^{1/2}\left(\boldsymbol{y} - \boldsymbol{D}_c\hat{\boldsymbol{\alpha}}_c\right)\right\|_2$, $\boldsymbol{D}_c$ is the sub-dictionary associated with class c, $\hat{\boldsymbol{\alpha}}_c$ is the final sub-coding vector associated with class c, and $\boldsymbol{W}_{final}$ is the final weight matrix.

5.3.2 The convergence of IR^{3}C

Eq. (5-21) is a local approximation of the RRC in Eq. (5-8), and in each iteration the objective function of Eq. (5-8) decreases by the IR^{3}C algorithm, i.e., in steps 3 and 4, $\boldsymbol{\alpha}^{(t)}$ can make $\sum_{i=1}^{n}\rho_{\boldsymbol{\theta}}\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}^{(t)}\right) + \sum_{j=1}^{m}\rho_o\left(\alpha_j^{(t)}\right) < \sum_{i=1}^{n}\rho_{\boldsymbol{\theta}}\left(y_i - \boldsymbol{r}_i\boldsymbol{\alpha}^{(t-1)}\right) + \sum_{j=1}^{m}\rho_o\left(\alpha_j^{(t-1)}\right)$. Since the cost function of Eq. (5-8) is lower bounded (≥0), the iterative minimization procedure in IR^{3}C will converge. Specifically, we stop the iteration if the following holds:

$$\left\|\boldsymbol{W}^{(t+1)} - \boldsymbol{W}^{(t)}\right\|_2 \Big/ \left\|\boldsymbol{W}^{(t)}\right\|_2 < \delta_{\boldsymbol{W}} \tag{5-23}$$

where $\delta_{\boldsymbol{W}}$ is a small positive scalar.

5.3.3 Complexity analysis

Generally speaking, the complexity of IR^{3}C and SRC [102] mainly lies in the coding process, i.e., Eq. (5-18) or (5-19) for IR^{3}C and Eq. (5-1) or Eq. (5-2) for SRC. It is known that the l_1-minimization, such as Eq. (5-1) for SRC, has a computational complexity of

$O(n^2m^{1.5})$ [228], where n is the dimensionality of face feature, and m is the number of dictionary atoms. It is also reported that the commonly used l_1-minmization solvers, e.g., l_1_magic [226] and l_1_*ls* [204], have an empirical complexity of $O(n^2m^{1.3})$ [204].

For IR^{3}C with β=2, the coding (i.e., Eq. (5-18)) is an l_2-regularized least square problem. The solution $\hat{\boldsymbol{\alpha}} = \left(\boldsymbol{D}^T\boldsymbol{W}\boldsymbol{D} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{D}^T\boldsymbol{W}\boldsymbol{y}$ could be got by solving $\left(\boldsymbol{D}^T\boldsymbol{W}\boldsymbol{D} + \lambda\boldsymbol{I}\right)\hat{\boldsymbol{\alpha}} = \boldsymbol{D}^T\boldsymbol{W}\boldsymbol{y}$ efficiently via conjugate gradient method [248], whose time complexity is about $O(k_1nm)$ (here k_1 is the iteration number in conjugate gradient method). Suppose that t iterations are used in IR^{3}C to update $\boldsymbol{W}$, the overall complexity of IR^{3}C with β=2 is about $O(tk_1nm)$. Usually t is less than 15. It is easy to see that IR^{3}C with β=2 has much lower complexity than SRC.

For IR^{3}C with β=1, the coding in Eq. (5-19) is an l_1-norm sparse coding problem, which could also be solved via conjugate gradient method. The complexity of IR^{3}C with β=1 will be about $O(tk_1k_2nm)$, where k_2 is the number of iteration to update $\boldsymbol{V}$. By experience, k_1 is less than 30 and k_2 is less 20, and then k_2k_1 is basically in the similar order to n. Thus the complexity of IR^{3}C with β=1 is about $O(tn^2m)$. Compared with SRC in case of FR without occlusion, although IR^{3}C needs several iterations (usually t=2) to update $\boldsymbol{W}$, its time consumption is still lower than or comparable to SRC. In FR with occlusion or corruption, for IR^{3}C usually t=15. In this case, however, SRC's complexity is $O(n^2(m+n)^{1.3})$ because it needs to use an identity matrix to code the occluded or corrupted pixels, as shown in Eq. (5-2). It is easy to conclude that IR^{3}C with β=1 has much lower complexity than SRC for FR with occlusion.

Although many faster l_1-norm minimization methods than l_1_magic [226] and l_1_*ls* [204] have been proposed recently, as reviewed in [126], by adopting them in SRC the running time is still larger than or comparable to the proposed IR^{3}C, as demonstrated in Section 5.4.5. In addition, in the iteration of IR^{3}C we can delete the element y_i that has very small weight because this implies that y_i is an outlier. Thus the complexity of IR^{3}C

can be further reduced. For example, in FR with real disguise on the AR database, about 30% pixels could be deleted.

5.4 Experimental Results

We perform experiments on benchmark face databases to demonstrate the performance of RRC. In Section 5.4.1, we give the parameter setting of RRC; in Section 5.4.2, we test RRC for FR without occlusion; in Section 5.4.3, we demonstrate the robustness of RRC to FR with random pixel corruption, random block occlusion and real disguise; in Section 5.4.4, the experiments on rejecting invalid testing images are performed. In Section 5.4.5, the running time is presented. Finally, some discussions of parameter selection are given in Section 5.4.6.

All the face images are cropped and aligned by using the locations of eyes. We normalize the testing image (or feature) and training image (or feature) to have unit l_2-norm energy. For AR [212] and Extended Yale B [99, 206] databases, the eye locations are provided by the databases. For Multi-PIE [213] database, we manually locate the eyes for the experiments in Section 5.4.2, and automatically detect the facial region by the face detector [249] for the experiments in Sections 5.4.4. In all experiments, the training samples are used as the dictionary $\boldsymbol{D}$ in coding. We denote by RRC_L_1 our RRC model with l_1-norm coefficient constraint (i.e., Eq. (5-19)), and by RRC_L_2 our RRC model with l_2-norm coefficient constraint (i.e., Eq. (5-18)). Both RRC_L_1 and RRC_L_2 are implemented by the IR^{3}C algorithm described in Section 5.3.1.

5.4.1 Parameter setting

In the weight function Eq. (5-14), there are two parameters, δ and μ, which need to be calculated in Step 2 of the IR^{3}C algorithm. δ is the parameter of demarcation point. When

the square of residual is larger than δ, the weight will be less than 0.5. To make the model robust to outliers, we compute δ as follows. Let $l=\lfloor \tau n \rfloor$, where scalar $\tau \in (0,1)$, and $\lfloor \tau n \rfloor$ outputs the largest integer smaller than τn. We set δ as

$$\delta = \psi_1(\boldsymbol{e})_l \tag{5-24}$$

where for a vector $\boldsymbol{e} \in \Re^n$, $\psi_1(\boldsymbol{e})_k$ is the k^{th} largest element of the set $\{ e_j^2, j=1,\ldots,n\}$.

Parameter μ controls the decreasing rate of weight $\boldsymbol{W}_{i,i}$. Here we simply let $\mu=\varsigma/\delta$, where $\varsigma=8$ is set as a constant. In the experiments, τ is fixed as 0.8 for FR without occlusion, and 0.6 for FR with occlusion. In addition, the regularization parameter λ in Eq. (5-18) or Eq. (5-19) is set as 0.001 by default.

For RRC_L$_1$, there is a parameter ε in updating the weight matrix $\boldsymbol{V}$: $\boldsymbol{V}_{j,j}^{(k+1)} = \lambda \left| (\alpha_j^{(k)})^2 + \varepsilon^2 \right|^{-1/2}$. According to [235], we choose ε as

$$\varepsilon^{(k+1)} = \min\left(\varepsilon^{(k)}, \psi_2\left(\boldsymbol{\alpha}^{(k)}\right)_L \Big/ m \right) \tag{5-25}$$

where for a vector $\boldsymbol{\alpha} \in \Re^m$, $\psi_2(\boldsymbol{\alpha})_i$ is the i^{th} largest element of the set $\{ |\alpha_j|, j=1,\cdots,m \}$. We set $L = \lfloor 0.01m \rfloor$. The above design of ε could not only make the numerical computing of weight $\boldsymbol{V}$ stable, but also ensure the iteratively reweighted least square achieve a sparse solution ($\varepsilon^{(k+1)}$ decreases to zero as k increases).

5.4.2 Face recognition without occlusion

We first validate the performance of RRC in FR with variations such as illumination and expression changes but without occlusion. We compare RRC with SRC [102], locality-constrained linear coding (LLC) [237], linear regression for classification (LRC) [101] and the benchmark methods such as nearest neighbor (NN), nearest feature line (NFL) [96] and linear support vector machine (SVM). In the experiments, PCA is used to reduce the dimensionality of original face images, and the Eigenface features are used for all the competing methods. Denote by $\boldsymbol{P}$ the PCA projection matrix, the step 3 of IR^{3}C

becomes:

$$\boldsymbol{\alpha}^{*} = \arg\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \left\| \boldsymbol{P}(\boldsymbol{W}^{(t)})^{1/2} \left(\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha} \right) \right\|_2^2 + \sum_{j=1}^{m} \rho_o \left(\alpha_j \right) \right\} \tag{5-26}$$

4.2.1) Extended Yale B Database: The Extended Yale B [99, 206] 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 5.2 shows the recognition rates versus feature dimension by NN, NFL, SVM, SRC, LRC, LLC and RRC methods. RRC_L_1 achieves better results than the other methods in all dimensions except that it is slightly worse than SVM and LLC when the dimension is 30. RRC_L_2 is better than SRC, LRC, LLC, SVM, NFL and NN when the dimension is 150 or higher. The best recognition rates of SVM, SRC, LRC, LLC, RRC_L_2 and RRC_L_1 are 97.0%, 98.3%, 96.0%, 97.6%, 98.9% and 99.8% respectively.

Table 5.2: Face recognition rates on the Extended Yale B database.

Dimension	30	84	150	300
NN	66.3%	85.8%	90.0%	91.6%
SVM	**92.4%**	94.9%	96.4%	97.0%
LRC	63.6%	94.5%	95.1%	96.0%
NFL	89.6%	94.1%	94.5%	94.9%
SRC	90.9%	95.5%	96.8%	98.3%
LLC	92.1%	96.4%	97.0%	97.6%
RRC_L_2	71.6%	94.4%	97.6%	98.9%
RRC_L_1	91.3%	**98.0%**	**98.8%**	**99.8%**

4.2.2) AR Database: As in [102], a subset (with only illumination and expression changes) that contains 50 male and 50 female subjects was chosen from the AR database [212] in this experiment. For each subject, the seven images from Session 1 were used for training, with other seven images from Session 2 for testing. The images were cropped to

60×43. The FR rates by the competing methods are listed in Table 5.3. We can see that apart from the case when the dimension is 30, RRC_L_1 achieves the highest rates among all methods, while RRC_L_2 is the second best. The reason that RRC works not very well with very low-dimensional feature is that the coding vector solved by Eq. (5-26) is not accurate enough to estimate $\boldsymbol{W}$ when the feature dimension is too low. Nevertheless, when the dimension is too low, all the methods cannot achieve good recognition rate. We can see that all methods achieve their maximal recognition rates at the dimension of 300, with 93.3% for SRC, 89.0% for LLC, 95.3% for RRC_L_2 and 96.3% for RRC_L_1.

Table 5.3: Face recognition rates on the AR database.

Dimension	30	54	120	300
NN	62.5%	68.0%	70.1%	71.3%
SVM	66.1%	69.4%	74.5%	75.4%
LRC	66.1%	70.1%	75.4%	76.0%
NFL	64.5%	69.2%	72.7%	73.4%
SRC	**73.5%**	83.3%	90.1%	93.3%
LLC	70.5%	80.7%	87.4%	89.0%
RRC_L_2	61.5%	84.3%	94.3%	95.3%
RRC_L_1	70.8%	**87.6%**	**94.7%**	**96.3%**

From Table 5.2 and Table 5.3, one can see that when the dimension of feature is not too low, RRC_L_2 could achieve similar performance to that of RRC_L_1, which implies that the l_1-sparsity constraint on the coding vector is not so important. This is because when the feature dimension is not too low, the dictionary (i.e., the feature set of the training samples) may not be over-complete enough, and hence using Laplacian to model the coding vector is not much better than using Gaussian. As a result, RRC_L_2 and RRC_L_1 will have similar recognition rates, but the former will have much less complexity.

4.2.3) Multi PIE database: The CMU Multi-PIE database [213] contains images of 337 subjects captured in four sessions with simultaneous variations in pose, expression, and illumination. Among these 337 subjects, all the 249 subjects in Session 1 were used

for training. To make the FR more challenging, four subsets with both illumination and expression variations in Sessions 1, 2 and 3, were used for testing. For the training set, as in [189], we used the 7 frontal images with extreme illuminations {0, 1, 7, 13, 14, 16, and 18} and neutral expression (refer to Fig. 5.3(a) for examples). For the testing set, 4 typical frontal images with illuminations {0, 2, 7, 13} and different expressions (smile in Sessions 1 and 3, squint and surprise in Session 2) were used (refer to Fig. 5.3(b) for examples with surprise in Session 2, Fig. 5.3(c) for examples with smile in Session 1, and Fig. 5.3(d) for examples with smile in Session 3). Here we used the Eigenface with dimensionality 300 as the face feature for sparse coding. Table 5.4 lists the recognition rates in four testing sets by the competing methods.

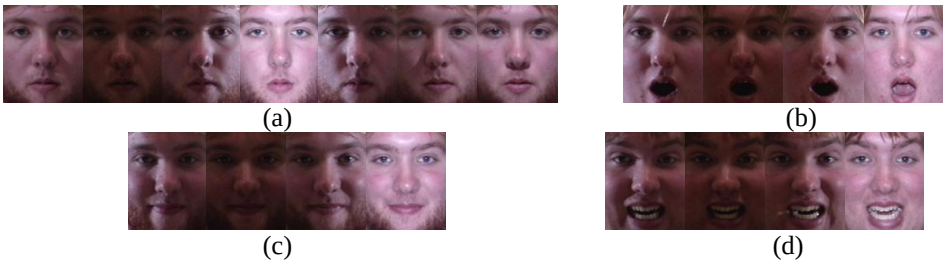


Figure 5.3: A subject in Multi-PIE database. (a) Training samples with only illumination variations. (b) Testing samples with surprise expression and illumination variations. (c) and (d) show the testing samples with smile expression and illumination variations in Session 1 and Session 3, respectively.

Table 5.4: Face recognition rates on Multi-PIE database. ('Smi-S1': set with smile in Session 1; 'Smi-S3': set with smile in Session 3; 'Sur-S2': set with surprise in Session 2; 'Squ-S2': set with squint in Session 2).

	Smi-S1	Smi-S3	Sur-S2	Squ-S2
NN	88.7%	47.3%	40.1%	49.6%
SVM	88.9%	46.3%	25.6%	47.7%
LRC	89.6%	48.8%	39.6%	51.2%
NFL	90.3%	50.0%	39.8%	52.9%
SRC	93.7%	60.3%	51.4%	58.1%
LLC	95.6%	62.5%	52.3%	64.0%
RRC_L_2	96.1%	70.2%	59.2%	58.1%
RRC_L_1	**97.8%**	**76.0%**	**68.8%**	**65.8%**

From Table 5.4, we can see that RRC_L_1 achieves the best performance in all tests, and RRC_L_2 performs the second best. Compared to the third best method, LLC, 6% and 2.3% improvements are achieved by RRC_L_1 and RRC_L_2, respectively. In addition, all the methods achieve their best results when Smi-S1 is used for testing because the training set is also from Session 1. From testing set Smi-S1 to Smi-S3, the variations increase because of the longer data acquisition time interval and the difference of smile (refer to Fig. 5.3(c) and Fig. 5.3(d)). The recognition rates of RRC_L_1 and RRC_L_2 drop by 21.8% and 25.9%, respectively, while those of NN, NFL, LRC, SVM, LLC and SRC drop by 41.4%, 40.3%, 40.8%, 42.6%, 33.1% and 33.4%, respectively. This validates that the RRC methods are much more robust to face variations than the other methods. Meanwhile, we could also see that FR with surprise and squint expression changes are much more difficult than FR with the smile expression change. In this experiment, the gap between RRC_L_2 and RRC_L_1 is relatively big. The reason is that the dictionary (size: 300×1743) used in this experiment is much over-complete and the 300-d eigenface feature doesn't contain enough discrimination, and thus the l_1-norm is much more powerful than the l_2-norm to regularize the representation of samples with big variations (e.g., expression changes).

5.4.3 Face recognition with occlusion

One of the most interesting features of sparse coding based FR in [102] is its robustness to face occlusion. In this section, we test the robustness of RRC to different kinds of occlusions, such as random pixel corruption, random block occlusion and real disguise. In the experiments of random corruption and random block occlusion, we compare RRC methods with SRC [102], LRC [101], Gabor-SRC [181] (only suitable for block occlusion) and correntropy-based sparse representation (CESR) [240], and NN is used as the baseline method. In the experiment of real disguise, we compare RRC with SRC, Gabor-SRC (GSRC) [181] (here GSRC refers to GRRC_L_1 in Chapter 4), CESR and other

state-of-the-art methods.

4.3.1) FR with pixel corruption: To be identical to the experimental settings in [102], we used Subsets 1 and 2 (717 images, normal-to-moderate lighting conditions) of the Extended Yale B database for training, and used Subset 3 (453 images, more extreme lighting conditions) for testing. The images were resized to 96×84 pixels. For each testing image, we replaced a certain percentage of its pixels by uniformly distributed random values within [0, 255]. The corrupted pixels were randomly chosen for each testing image and the locations are unknown to the algorithm.

Fig. 5.4 shows a representative example of RRC_L_1 and RRC_L_2 with 70% random corruption. Fig. 5.4(a) is the original sample, and Fig. 5.4(b) shows the testing image with random corruption. It can be seen that the corrupted face images are difficult to recognize, even for humans. The estimated weight maps of RRC_L_1 and RRC_L_2 are shown in the top and bottom rows of Fig. 5.4(c) respectively, from which we can see not only the corrupted pixels but also the pixels in the shadow region have low weights. Fig. 5.4(d) shows the coding coefficients of RRC_L_1 (top row) and RRC_L_2 (bottom row), while Fig. 5.4(e) shows the reconstructed images of RRC_L_1 (top row) and RRC_L_2 (bottom row). It can be seen that for RRC_L_1 only the dictionary atoms with the same label as the testing sample have big coefficients and the reconstructed image is faithful to the original image (Fig. 5.4(a)) but with better visual quality (the shadow which brings difficulties to recognition is removed). For RRC_L_2, although the coefficients are not sparse, the visual quality of the reconstructed image is also good and the classification performance is similar to RRC_L_1, which are shown in Table 5.5.

Table 5.5 shows the results of SRC, CESR, LRC, NN, RRC_L_2 and RRC_L_1 under different percentage of corrupted pixels. Since all competing methods could achieve no bad performance from 0% to 50% corruptions, we only list the average recognition rate for 0%~50% corruptions. One can see that when the percentage of corrupted pixels is between 0% and 50%, RRC_L_1, RRC_L_2, and SRC could correctly classify all the testing images. Surprisingly, CESR does not correctly recognize all the testing images in that

case. However, when the percentage of corrupted pixels is more than 70%, the advantage of RRC_L$_1$, RRC_L$_2$, and CESR over SRC is clear. Especially, RRC_L$_1$ achieves the best performance in all cases, with 100% (99.6% and 67.1%) in 70% (80% and 90%) corruption, while SRC only has a recognition rate of 90.7% (37.5% and 7.1%). LRC and NN are sensitive to the outliers, with much lower recognition rates than others. All RRC methods achieve better performance than CESR in all cases, which validates that the RRC model could suppress outliers better.

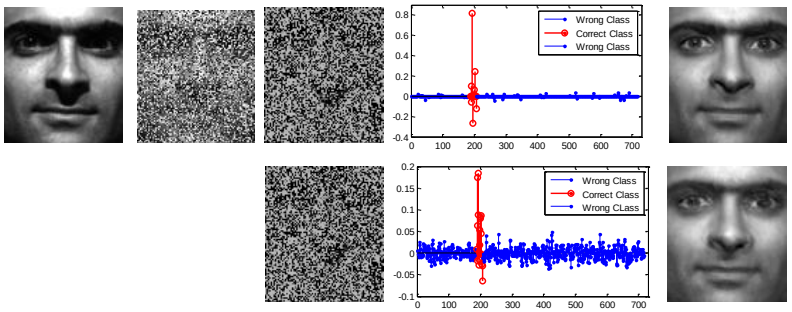


Figure 5.4: Recognition under random corruption. (a) Original image $\boldsymbol{y}_0$ from Extended Yale B database. (b) Testing image $\boldsymbol{y}$ with random corruption. (c) Estimated weight map of RRC_L$_1$ (top row) and RRC_L$_2$ (bottom). (d) Estimated representation coefficients $\boldsymbol{\alpha}$ of RRC_L$_1$ and RRC_L$_2$. (e) Reconstructed images $\boldsymbol{y}_{rec}$ of RRC_L$_1$ and RRC_L$_2$.

Table 5.5: The recognition rates of RRC, LRC, NN, SRC and CESR versus different percentage of corruption.

Corruption(%)	0~50	60	70	80	90
NN	89.3%	46.8%	25.4%	11.0%	4.6%
SRC [102]	**100%**	99.3%	90.7%	37.5%	7.1%
LRC	95.8%	50.3%	26.4%	9.9%	6.2%
CESR	97.4%	96.2%	97.8%	93.8%	41.5%
RRC_L$_2$	**100%**	**100%**	99.8%	97.8%	43.3%
RRC_L$_1$	**100%**	**100%**	**100%**	**99.6%**	**67.1%**

4.3.2) FR with block occlusion: In this section we test the robustness of RRC to block occlusion. We also used the same experimental settings as in [102], i.e., Subsets 1 and 2

of Extended Yale B for training, Subset 3 for testing, and replacing a randomly located square block of a testing image with an unrelated image, as illustrated in Fig. 5.5(b). The face images were resized to 96×84.

Fig. 5.5 shows an example of occluded face recognition (30% occlusion) by using RRC_L$_1$ and RRC_L$_2$. Fig. 5.5 (a) and (b) are the original sample from Extended Yale B database and the occluded testing sample. Fig. 5.5 (c) shows the estimated weight maps of RRC_L$_1$ (top row) and RRC_L$_2$ (bottom row), from which we could see that both of them assign big weights (e.g., 1) to the un-occluded pixels, and assign low weight (e.g., 0) to the occluded pixels. The estimated representation coefficients of RRC_L$_1$ and RRC_L$_2$ are shown in the top row and bottom row of Fig. 5.5 (d) respectively. It can be seen that RRC_L$_1$ could achieve very sparse coefficients with significant values on the atoms of correct class; the coefficients by RRC_L$_2$ also have significant values on the atoms of correct class but they are not sparse. From Fig. 5.5 (e), we see that both RRC_L$_1$ and RRC_L$_2$ have very good image reconstruction quality, effectively removing the block occlusion and the shadow.

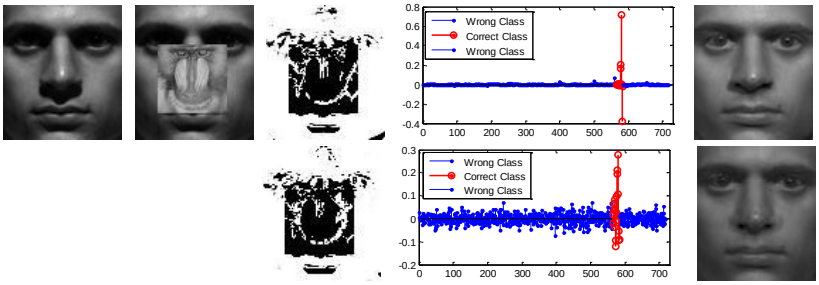


Figure 5.5: Recognition under 30% block occlusion. (a) Original image $\boldsymbol{y}_0$ from Extended Yale B. (b) Testing image $\boldsymbol{y}$ with random corruption. (c) Estimated weight maps of RRC_L$_1$ (top row) and RRC_L$_2$ (bottom row). (d) Estimated representation coefficients $\boldsymbol{\alpha}$ of RRC_L$_1$ and RRC_L$_2$. (e) Reconstructed images $\boldsymbol{y}_{rec}$ of RRC_L$_1$ and RRC_L$_2$.

Table 5.6 lists the detailed recognition rates of RRC_L$_1$, RRC_L$_2$, SRC, LRC, NN, GSRC and CESR under the occlusion percentage from 0% to 50%. From Table 5.6, we

see that RRC_L_1 has the best accuracy, and RRC methods achieve much higher recognition rates than SRC when the occlusion percentage is larger than 30% (e.g., more than 22% (6%) improvement at 50% (40%) occlusion). Compared to GSRC, RRC still gets better results without using the enhanced Gabor features. CESR gets worse results than SRC in this experiment. This may be because FR with block occlusion is more difficult than that of pixel corruption, but it shows that CESR could not accurately identify the outlier points in such block occlusion (i.e., outlier points have similar intensity as the face pixels). Encouragingly, RRC_L_2 has competing recognition rates with RRC_L_1 (even better than RRC_L_1 40% and 50% occlusion).

Table 5.6: The recognition rates of RRC, LRC, NN, GSRC, SRC and CESR under different levels of block occlusion.

Occlusion (%)	0	10	20	30	40	50
NN	94.0%	92.9%	85.4%	73.7%	62.9%	45.7%
SRC [102]	**100%**	**100%**	99.8%	98.5%	90.3%	65.3%
LRC	**100%**	**100%**	95.8%	81.0%	63.8%	44.8%
GSRC[181]	**100%**	**100%**	**100%**	**99.8%**	96.5%	87.4%
CESR	94.7%	92.7%	89.8%	83.9%	75.5%	57.4%
RRC_L_2	**100%**	**100%**	**100%**	**99.8%**	**97.6%**	**87.8%**
RRC_L_1	**100%**	**100%**	**100%**	**99.8%**	96.7%	87.4%

4.3.4) FR with real face disguise: A subset from the AR database is used in this experiment. This subset consists of 2,599 images from 100 subjects (26 samples per class except for a corrupted image w-027-14.bmp), 50 males and 50 females. We perform two tests: one follows the experimental settings in [102], while the other one is more challenging. The images were resized to 83×60 in the first test and 42×30 in the second test.

In the first test, 799 images (about 8 samples per subject) of non-occluded frontal views with various facial expressions in Sessions 1 and 2 were used for training, while two separate subsets (with sunglasses and scarf) of 200 images (1 sample per subject per Session, with neutral expression) for testing. Fig. 5.6 illustrates the classification process

of RRC_L_1 by using an example. Fig. 5.6(a) shows a testing image with sunglasses; Figs. 5.6(b) and 5.6(c) show the initialized and final weight maps, respectively; Fig. 5.6(d) shows one template image of the identified subject. The convergence of the IR^{3}C algorithm to solve the RRC model is shown in Fig. 5.6(e), and Fig. 5.6(f) shows the reconstruction error of each class, with the correct class having the lowest value.

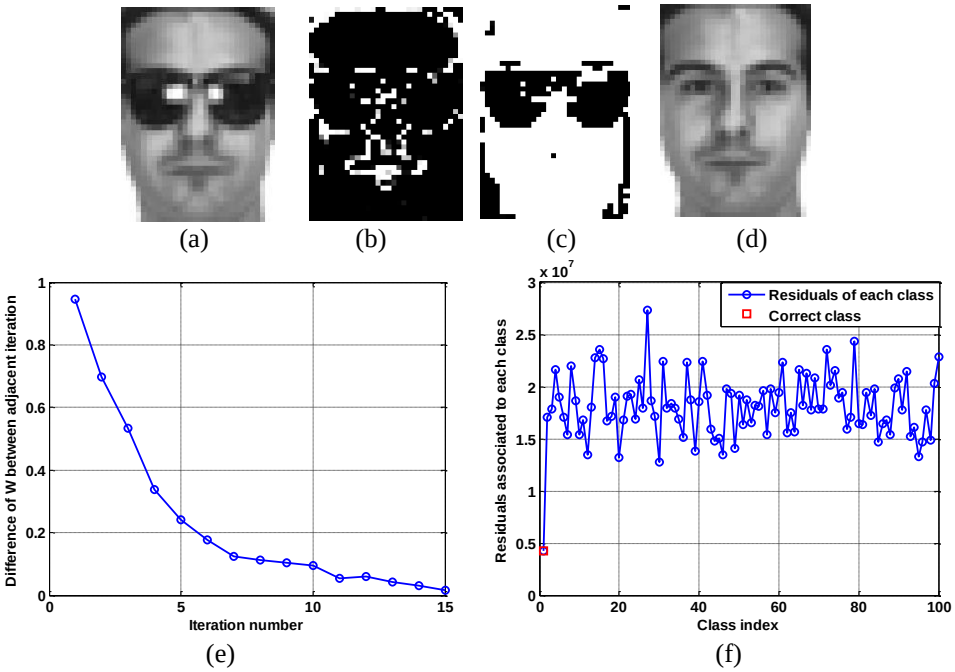


Figure 5.6: An example of face recognition with disguise using RRC_L_1. (a) A testing image with sunglasses. (b) The initialized weight map. (c) The weight map when IR^{3}C converges. (d) A template image of the identified subject. (e) The convergence curve of IR^{3}C. (f) The residuals of each class by RRC_L_1.

The FR results by the competing methods are listed in Table 5.7. We see that the RRC methods achieve much higher recognition rates than SRC, GSRC and CESR, while RRC_L_1 and RRC_L_2 achieve similar results. CESR has similar performance to RRC methods in FR with sunglass, but has much worse recognition rate in dealing with scarf. Similar to the case of FR with block occlusion, CESR is not robust enough for more challenging case (e.g., scarf covers about 40% face region). The proposed RRC methods also significantly outperform other state-of-the-art methods, including [250] with 84% on sunglasses and 93% on scarf, and [233] with 93% on sunglasses and 95.5% on scarf.

Table 5.7: Recognition rates by competing methods on the AR database with disguise occlusion.

Algorithms	Sunglasses	Scarves
SRC [102]	87.0%	59.5%
GSRC [181]	93%	79%
CESR	99%	42.0%
RRC_L_2	99.5%	96.5%
RRC_L_1	**100%**	**97.5%**

Table 5.8: Recognition rates by competing methods on the AR database with complex disguise occlusion.

Algorithms	Session 1		Session 2	
	Sunglasses	Scarves	Sunglasses	Scarves
SRC [102]	89.3%	32.3%	57.3%	12.7%
GSRC [181]	87.3%	85%	45%	66%
CESR	95.3%	38%	79%	20.7%
RRC_L_2	99.0%	**94.7%**	84.0%	**77.3%**
RRC_L_1	**99.0%**	93.3%	**89.3%**	76.3%

In the second test, we conduct FR with more complex disguise (disguise with variations of illumination and longer data acquisition interval). 400 images (4 neutral images with different illuminations per subject) of non-occluded frontal views in Session 1 were used for training, while the disguised images (3 images with various illuminations and sunglasses or scarves per subject per Session) in Sessions 1 and 2 for testing. Table 5.8 lists the results by competing methods. Clearly, the RRC methods achieve much better results than SRC, GSRC and CESR. Interestingly, CESR works well in the case of Sunglasses disguise but poor in the case of Scarves disguise, while GSRC the reverse. In addition, the average improvements of RRC_L_1 over SRC, GSRC and CESR are respectively 25.9%, 28% and 7% on sunglasses, and respectively 62.3%, 9.3% and 60.5% on scarf. In this experiment, RRC_L_1 is slightly better than RRC_L_2 on sunglasses, with RRC_L_2 slightly better than RRC_L_1 on scarf.

5.4.4 Face validation

In practical FR systems, it is important to reject invalid face images which have no template in the database. It should be noted that "*rejecting invalid images not in the entire database is much more difficult than deciding if two face images are the same subject*" [189]. In this section we check whether the proposed RRC methods could have good face validation performance. Similar to [102, 189], all the competing methods use the *Sparsity Concentration Index* (SCI) proposed in [102] to do face validation with the coding coefficient. Like [189], we used the large-scale Multi-PIE face database to perform face validation experiments. All the 249 subjects in Session 1 were used as the training set, with the same subjects in Session 2 as customer images. The remaining 88 subjects (37 subjects with ID between 251 and 292 from Session 2 and 51 subjects with ID between 293 and 346 from Session 3) different from the training set were used as the imposter images. For the training set, as in [189] we used the 7 frontal images with extreme illuminations {0, 1, 7, 13, 14, 16, and 18} and neutral expression (refer to Fig. 5.3(a) for examples). For the testing set, 10 typical frontal images of illuminations {0, 2, 4, 6, 8, 10, 12, 14, 16, 18} taken with neutral expressions were used. In this experiment, the testing face images were automatically detected by using Viola and Jones' face detector [249] and then automatically aligned to the size of 60×48 without manual intervention (a testing image is automatically aligned to the training data of each subject by the method in [189]).

Fig. 5.7 plots the ROC (receiver operating characteristic) curves of the competing methods: SRC, RRC_L_1, RRC_L_2 and CESR. It can be seen that CESR works the worst while RRC_L_2 works the best. For instance, when the false positive rate is 0.1, the true positive rate is 82.6% for CESR, 90.7% for SRC, 93.3% for RRC_L_1 and 95.8% for RRC_L_2. It is a little surprising that RRC_L_2 with l_2-norm coefficient constraint achieves better face validation results than the l_1-norm coefficient constrained methods, e.g., SRC, RRC_L_1, and much better than CESR. The reason may be that the l_1-norm constraint, especially the nonnegative sparse constraint (for CESR), which strongly forces the coding

coefficients to be sparse, will force one specific class to represent the input invalid testing sample, and hence incorrectly recognize this testing sample. Comparatively, l_2-norm constraint does not force the coding coefficients to be sparse, which allows the representation coefficients of invalid testing samples to be evenly distributed across different classes. Therefore the incorrect recognition can be avoided. In addition, RRC_L_1 are better than SRC and CESR, validating that the signal fidelity term of RRC_L_1 is more robust.

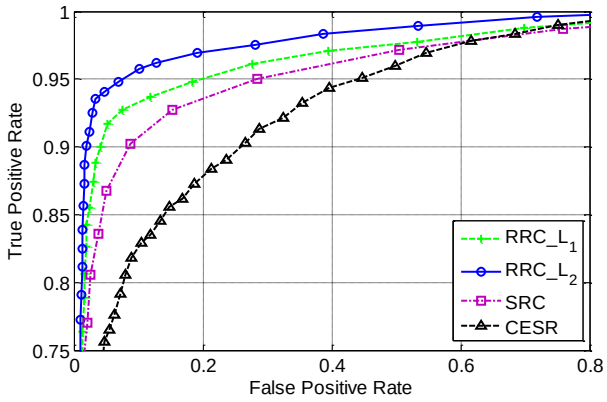
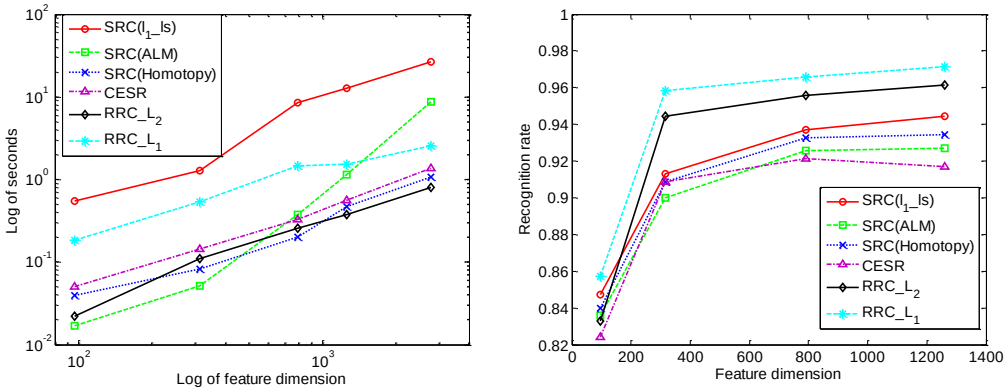


Figure 5.7: Subject validation on the large-scale Multi PIE.

5.4.5 Running time comparison

Apart from recognition rate, computational expense is also an important issue for practical FR systems. In this section, the running time of the competing methods, including SRC, GSRC, CESR, RRC_L_2 and RRC_L_1, is evaluated using two FR experiments (without occlusion and with real disguise). The programming environment is Matlab version 7.0a. The desktop used is of 3.16 GHz CPU and with 3.25G RAM. All the methods are implemented using the codes provided by the authors. For SRC, we adopt l_1_*ls* [204], and two fast l_1-minimization solvers, ALM and Homotopy [126], to implement the sparse coding step.



(a) (b)

Figure 5.8: Running time and recognition rates by the competing methods under different feature dimension in FR without occlusion.

The first experiment is FR without occlusion on the AR database, whose experimental setting is the same as that in Section 5.4.2 but with various down-sampled face features (i.e., 12×8, 21×15, 33×24, 42×30 and 126×90). Fig. 5.8 compares the running time (Fig. 5.8 (a)) and recognition rates (Fig. 5.8 (b)) of the competing methods under various feature dimensions. From Fig. 5.8 (a), it can be seen that RRC_L_2, CESR and SRC (ALM) have obvious faster speed than other methods. RRC_L_1 is also much more efficient than SRC (l_1_*ls*), the slowest one.

With the feature of 792 (33×24) dimensions, RRC_L_2, CESR, RRC_L_1, SRC (l_1_*ls*), SRC (ALM) and SRC (Homotopy) take 0.257, 0.330, 1.450, 8.551, 0.377 and 0.199 seconds, respectively. RRC_L_1 achieves the best recognition rates followed by RRC_L_2, as shown in Fig. 5.8(b). Although CESR is also fast, its recognition rates are lower than other methods. It can be concluded that compared to SRC and CESR, RRC_L_2 has good recognition rate with much less or comparable computation expense, while RRC_L_1 has much higher recognition rate.

The second experiment is FR with real face disguise. The experimental settings are described in Section 5.4.3. The dictionary has 800 training samples with size 83×60 in Test 1, and 400 training samples with size 42×30 in Test 2. The recognition rates have been reported in Table 5.7 (for Test 1) and Table 5.8 (for Test 2). Table 5.9 lists the

average computational expense and recognition rates of different methods on Test1 and Test2. Clearly, RRC_L_2 has the least computation time, followed by CESR and RRC_L_1. SRC has rather high computation burden even with fast solvers such as ALM and Homotopy, which is because an additional identity matrix is utilized to code occlusion. For the recognition rate, SRC's performance is the worst, and CESR also has rather bad recognition rate in FR with scarf in each test. GSRC solved by l_1_*ls* has lower time cost than SRC (l_1_*ls*) but still very slow. Considering both the recognition rate and running time, RRC_L_1 and RRC_L_2 are the best ones. RRC_L_1 gets the highest recognition rates in all case, at the same time with faster speed than SRC and GSRC. RRC_L_2 is the fastest one in all case, at the same time with the second best performance (e.g., in the Test 2 of FR with scarf, 63.5%, 10.5% and 56.6% higher than SRC(l_1_*ls*), GSRC, and CESR in average).

Table 5.9: The average running time (seconds) of competing methods in FR with real face disguise. The values in parenthesis are the average recognition rate.

Method	Test 1-sunglass	Test 1-scarf	Test 2-sunglass	Test 2- scarf
CESR[240]	2.5 (99.0%)	3.6 (42.0%)	0.5 (87.2%)	0.5 (29.4%)
SRC(l_1_*ls*)	662.1 (87.0%)	727.1 (59.5%)	38.2 (73.3%)	47.7 (22.5%)
SRC(ALM)	36.0 (84.5%)	36.4 (58.5%)	2.3 (72.4%)	2.4 (21.7%)
SRC(Homotopy)	14.0 (65.0%)	13.7 (37.5%)	3.6 (60.0%)	3.6 (17.3%)
GSRC[181]	119.3 (93.0%)	118.1 (79.0%)	13.0 (66.2%)	12.5 (75.5%)
RRC_L_1	8.7 (**100%**)	8.6 (**97.5%**)	2.1 (**94.2%**)	2.0 (84.8%)
RRC_L_2	**2.2** (99.5%)	**2.0** (96.5%)	**0.2** (91.5%)	**0.2** (**86.0%**)

5.4.6 Parameter discussion

In this section, we discuss the effect of parameter δ in RRC on the final recognition rate. As described below Eq. (5-14) and in Section 5.4.1, the parameter δ is a key parameter to distinguish inliers or outliers (if the residual's square of a pixel is larger than δ, its weight will be less than 0.5; otherwise, its weight is bigger than 0.5). In our implementation, we use the parameter τ to estimate δ, as described in Eq. (5-24). Hence, it is necessary to

discuss the selection of τ. Here we take the experiment with various level random pixel corruption (experimental settings are described in Section 5.4.3) as an example to discuss the selection of τ for RRC. Fig. 5.9 plots the recognition rates of RRC_L_1 versus different values of τ for 0%, 30%, 60%, and 90% pixel corruption. It can be seen that for moderate corruption (i.e., 0%~60%), RRC_L_1 could get very good performance (i.e., more than 95%) in a broad range of τ. For all percentages of pixel corruption, the best performance could be achieved when τ=0.5. Compared to CESR [240], whose kernel size is very sensitive to the corruption percentage (please refer to Section 5.7 of [240]), our proposed RRC method is easy to tune and is more robust to occlusion. Usually the domain of τ could be set as [0.5, 0.8]. It is reasonable because at least 50% samples should be trusted when there are large percent of outliers.

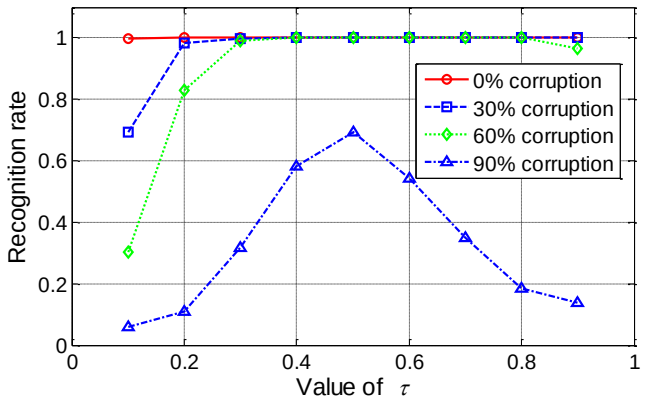


Figure 5.9: Recognition performance versus τ in estimating δ of RRC's weight function.

5.4 Summary

This Chapter presented a novel robust regularized coding (RRC) model and an associated effective iteratively reweighted regularized robust coding (IR^{3}C) algorithm for robust face recognition (FR). One important advantage of RRC is its robustness to various types of outliers (e.g., occlusion, corruption, expression, etc.) by seeking for an approximate MAP (maximum a posterior estimation) solution of the coding problem. By assigning adaptively

and iteratively the weights to the pixels according to their coding residuals, the IR^{3}C algorithm could robustly identify the outliers and reduce their effects on the coding process. The proposed RRC methods were extensively evaluated on FR with different conditions, including variations of illumination, expression, occlusion, corruption, and face validation. The experimental results clearly demonstrated that RRC outperforms significantly previous state-of-the-art methods, such as SRC, CESR and GSRC. In particular, RRC with l_2-norm regularization could achieve very high recognition rate but with low computational cost, which makes it a very good candidate scheme for practical robust FR systems.

Chapter 6. Fisher Discrimination Dictionary Learning for Sparse Representation

6.1 Introduction

As reviewed in Sections 1.3 and 2.3, the choice of the dictionary that sparsely represents the signals is crucial for the success of sparse representation modeling, and learning dictionary from training data has led to state-of-the-art results in many practical applications, including pattern classification. In sparsity based classification, usually there are two phases: coding (or representation) and classification. In the first phase, the testing signal/image is sparsely coded over a dictionary of atoms, and in the second phase, the classification is performed based on the coding coefficients and the dictionary. The dictionary for sparse coding could be predefined. For example, Wright *et al.* [102] directly used the training samples of all classes as the dictionary to code the testing face image, and then classified the testing face image by evaluating which class will lead to the minimal reconstruction error. Although this sparse representation based classification (SRC) scheme has shown interesting FR results, the dictionary used in it may not be effective enough to represent the testing images due to the uncertain and noisy information in the original training images. On the other hand, the number of atoms of such a dictionary can be very big, which increases the coding complexity. In addition, dictionary learning could remove unuseful information and introduce appropriate regularizations, which could well exploit the discriminative information hidden in the training samples. Furthermore, selecting a subset of the analytically designed off-the-shelf bases as dictionary (e.g., [144] uses Haar wavelet bases and Gabor bases as the dictionary) might be universal to all types of images but will not be effective enough for a specified type of images such as face, digit and texture images. In fact, all the above mentioned

problems of predefined dictionary for sparse representation can be addressed, at least to some extent, by learning properly a non-parametric dictionary from the original training samples.

Dictionary learning (DL) aims to learn from the training samples the sparse domain where the given signal could be sparsely coded for processing. Many DL methods have been proposed for image processing [129, 131, 151] and classification [145, 155-159, 162, 184, 196, 251]. One representative DL method for image processing is the KSVD algorithm [129], which learns an over-complete dictionary of atoms from a training dataset of natural image patches. However, KSVD is not suitable for classification tasks because it only requires that the learnt dictionary should faithfully represent the training samples.

As reviewed in Section 2.3, the dictionary learning model for classification tasks usually need to introduce additional priors. One important way is to require the coding coefficient have discrimination in the phase of dictionary learning, such as supervised dictionary learning [156], discriminative K-SVD [157], Label-Consistent K-SVD [184] and joint learning and dictionary construction [155]. All the works in [155-157, 184] try to learn a common dictionary shared by all classes, as well as a classifier of coefficients for classification. However, the shared dictionary loses the correspondence between the dictionary atoms and the class labels, and hence performing classification based on the reconstruction error associated with each class is not allowed. Another direction is to learn a structure dictionary whose atoms have correspondence to the object classes [158, 162]. In this case the dictionary atoms of one class are required to be able to well reconstruct the training samples of the same class, but have poor representation ability to other classes. Most of the previous methods [158, 162] of this kind use only the reconstruction error associated with each class as the discriminative information for classification, but they do not enforce discriminative information into the sparse coding coefficients in dictionary learning and final classification.

In this Chapter we propose a new discriminative DL framework which employs the

Fisher discrimination criterion to learn a structured dictionary (i.e. the dictionary atoms have correspondence to the class labels so that the reconstruction error associated with each class can be used for classification). Meanwhile, the Fisher discrimination criterion is imposed on the coding coefficients to make them discriminative. To this end, in the DL process we make the sparse coding coefficients have small within-class scatter but big between-class scatter, and at the same time we make each class-specific sub-dictionary in the whole structured dictionary have good representation ability to the training samples from the associated class but poor representation ability for other classes. With the proposed Fisher discrimination based DL (FDDL) method, both the reconstruction error and the coding coefficient will be discriminative, and hence a new classification scheme is proposed to exploit such information. The extensive experiments on the application of face recognition, digit recognition, gender classification and object categorization show that better or very competitive performance could be achieved by FDDL compared to the state-of-the-art methods.

6.2 Some Related Works

In this section, we briefly review some DL methods which are closely related to our proposed FDDL. One characteristic of this kind of learnt dictionary is that the dictionary atoms have class labels in correspondence to the object classes. Therefore, when the sparse coding coefficient (i.e., $\boldsymbol{\alpha}=[\boldsymbol{\alpha}_1;\boldsymbol{\alpha}_2;\cdots;\boldsymbol{\alpha}_K]$ where $\boldsymbol{\alpha}_i$ is the coefficient vector associated with class *i*) of a testing sample $\boldsymbol{y}$ is computed, the class-specific representation residual (i.e., $\|\boldsymbol{y}-\boldsymbol{D}_i\boldsymbol{\alpha}_i\|_2$ where $\boldsymbol{D}_i$ is the sub-dictionary associated with class *i*) could be used to do the final classification.

Denote by $\boldsymbol{A}_i$ as the training samples of the i^{th} object class, with each column of $\boldsymbol{A}_i$ being a training sample vector. The class-specific dictionary $\boldsymbol{D}_i=[\boldsymbol{d}_1,\boldsymbol{d}_2,...,\boldsymbol{d}_{p_i}]\in\Re^{m\times p_i}$

could be learnt class by class:

$$\min_{\boldsymbol{D}_i,\boldsymbol{Z}_i}\left\{\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{Z}_i\right\|_F^2+\lambda\left\|\boldsymbol{Z}_i\right\|_1\right\}\quad \text{s.t.}\quad \boldsymbol{d}_j^T\boldsymbol{d}_j=1,\forall j \tag{6-1}$$

where $\boldsymbol{Z}_i$ is the representation coefficient matrix of $\boldsymbol{A}_i$ on $\boldsymbol{D}_i$.

Eq. (6-1) is the basic model of learning dictionary with labels. Although it seems similar to the model of KSVD [129], note that here the dictionary is trained from the samples of a specific class. Therefore, DL by Eq. (6-1) is much more suitable for classification task than directly applying KSVD to all the training samples.

The metaface learning method [251] adopts the model in Eq. (6-1) to train dictionaries for FR. However, metaface learning does not introduce into the learnt dictionary more discrimination information, which is very crucial for classification tasks. Unlike metafaces learning that trains the class-specific dictionary separately, Ramirez *et al.* [158] used an incoherence promoting term to encourage the dictionaries associated with different classes as independent as possible. The so-called dictionary learning with structured incoherence (DLSI) could be formulated as

$$\min_{\{\boldsymbol{D}_i,\boldsymbol{Z}_i\},i=1,\cdots,K}\sum_{i=1}^{K}\left\{\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{Z}_i\right\|_F^2+\lambda\left\|\boldsymbol{Z}_i\right\|_1\right\}+\eta\sum_{i\neq j}\left\|\boldsymbol{D}_i^T\boldsymbol{D}_j\right\|_F^2\quad \text{s.t.}\quad \boldsymbol{d}_k^T\boldsymbol{d}_k=1,\forall k \tag{6-2}$$

where term $\sum_{i\neq j}\left\|\boldsymbol{D}_i^T\boldsymbol{D}_j\right\|_F^2$ is to promote incoherence between the different dictionaries. It is easy to see that the dictionary incoherence term could make the class-specific dictionary more distinctive and hence benefit the final classification.

6.3 Fisher Discrimination Dictionary Learning (FDDL)

To improve the performance of SRC [102] and previous DL methods [155-158, 162, 184, 196], we propose here a novel Fisher discrimination based DL (FDDL) scheme. Instead of learning a shared dictionary to all classes, we aim to learn a structured dictionary $\boldsymbol{D}=[\boldsymbol{D}_1,\boldsymbol{D}_2,\ldots,\boldsymbol{D}_K]$, where $\boldsymbol{D}_i$ is the class-specified sub-dictionary associated with the i^{th}

class, and K is the total number of classes. With this structured dictionary, we could use the reconstruction error associated with each class for classification, as in the original SRC method [102].

Denote by $\boldsymbol{A}=[\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_K]$ the set of original training samples, where $\boldsymbol{A}_i$ is the sub-set of the training samples from class i. Denote by $\boldsymbol{X}$ the sparse coding coefficient matrix of $\boldsymbol{A}$ over $\boldsymbol{D}$, i.e. $\boldsymbol{A}\approx\boldsymbol{D}\boldsymbol{X}$. We can write $\boldsymbol{X}$ as $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K]$, where $\boldsymbol{X}_i$ is the sub-matrix containing the coding coefficients of $\boldsymbol{A}_i$ over $\boldsymbol{D}$. Apart from requiring that the dictionary $\boldsymbol{D}$ should have powerful reconstruction capability of $\boldsymbol{A}$ (i.e. $\boldsymbol{A}\approx\boldsymbol{D}\boldsymbol{X}$), we also require that $\boldsymbol{D}$ should have powerful discriminative capability of images in $\boldsymbol{A}$. To this end, we propose the following FDDL model:

$$\min_{\boldsymbol{D},\boldsymbol{X}} \left\{ r(\boldsymbol{A},\boldsymbol{D},\boldsymbol{X}) + \lambda_1 \|\boldsymbol{X}\|_1 + \lambda_2 f(\boldsymbol{X}) \right\} \tag{6-3}$$

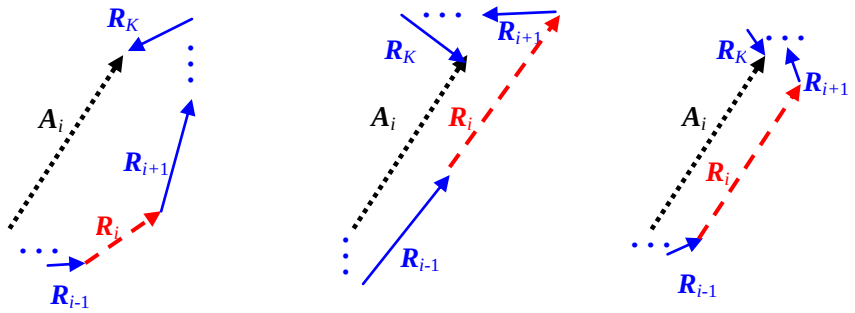
where $r(\boldsymbol{A},\boldsymbol{D},\boldsymbol{X})$ is the discriminative data fidelity term; $\|\boldsymbol{X}\|_1$ is the sparsity constraint; $f(\boldsymbol{X})$ is a discrimination constraint imposed on the coefficient matrix $\boldsymbol{X}$; and λ_1 and λ_2 are scalar parameters. Next let's discuss the design of $r(\boldsymbol{A},\boldsymbol{D},\boldsymbol{X})$ and $f(\boldsymbol{X})$ based on the Fisher discrimination criterion.

6.3.1 Discriminative data fidelity term $r(\boldsymbol{A},\boldsymbol{D},\boldsymbol{X})$

We can write $\boldsymbol{X}_i$, the representation of $\boldsymbol{A}_i$ over $\boldsymbol{D}$, as $\boldsymbol{X}_i = [\boldsymbol{X}_i^1; \ldots; \boldsymbol{X}_i^j; \ldots; \boldsymbol{X}_i^K]$, where $\boldsymbol{X}_i^j$ is the coding coefficients of $\boldsymbol{A}_i$ over the sub-dictionary $\boldsymbol{D}_j$. Denote the representation of $\boldsymbol{D}_k$ to $\boldsymbol{A}_i$ as $\boldsymbol{R}_k=\boldsymbol{D}_k\boldsymbol{X}_i^k$. First of all, the dictionary $\boldsymbol{D}$ should be able to well represent $\boldsymbol{A}_i$, and there is $\boldsymbol{A}_i\approx\boldsymbol{D}\boldsymbol{X}_i=\boldsymbol{D}_1\boldsymbol{X}_i^1+\ldots+\boldsymbol{D}_i\boldsymbol{X}_i^i+\ldots+\boldsymbol{D}_K\boldsymbol{X}_i^K=\boldsymbol{R}_1+\ldots+\boldsymbol{R}_i+\ldots+\boldsymbol{R}_K$. Second, since $\boldsymbol{D}_i$ is required to be associated with the i^{th} class, it is expected that $\boldsymbol{A}_i$ should be well represented by $\boldsymbol{D}_i$ but not by $\boldsymbol{D}_j$, $j\neq i$. This implies that $\boldsymbol{X}_i^i$ should have some significant coefficients such that $\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2$ is small, while $\boldsymbol{X}_i^j$ should have nearly zero coefficients such that $\|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2$ is small. Thus we can define the discriminative data fidelity term as

$$r(\boldsymbol{A}_i, \boldsymbol{D}, \boldsymbol{X}_i) = \|\boldsymbol{A}_i - \boldsymbol{D}\boldsymbol{X}_i\|_F^2 + \|\boldsymbol{A}_i - \boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2 + \sum_{\substack{j=1 \\ j\neq i}}^{K} \|\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F^2 \quad (6\text{-}4)$$

An intuitive explanation of three terms in $r(\boldsymbol{A}_i,\boldsymbol{D},\boldsymbol{X}_i)$ is shown in Fig. 6.1. Fig. 6.1(a) shows that although $\boldsymbol{D}$ is ensured to represent $\boldsymbol{A}_i$ well, $\boldsymbol{R}_i$ may deviate much from $\boldsymbol{A}_i$ so that $\boldsymbol{D}_i$ could not well represent $\boldsymbol{A}_i$. If we add another constraint that $\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\|_F^2$ is small, better discrimination will be achieved, as shown in Fig. 6.1(b). Nonetheless, $\boldsymbol{A}_i$ may also be well represented by other sub-dictionaries, e.g. $\boldsymbol{D}_{i-1}$ in Fig. 6.1(b), which reduces the discrimination capability of $\boldsymbol{D}$. With the third constraint that the representation of $\boldsymbol{D}_j$, $j\neq i$, to $\boldsymbol{A}_i$ is small, the proposed discriminative fidelity term could overcome this problem, as shown in Fig.6.1(c).



(a) (b) (c)

Figure 6.1: Illustration of the fidelity constraints. (a) Only $\boldsymbol{D}$ is required to well represent $\boldsymbol{A}_i$. (b) Both $\boldsymbol{D}$ and $\boldsymbol{D}_i$ are required to well represent $\boldsymbol{A}_i$. (c) The proposed discriminative fidelity term in Eq. (6-4).

6.3.2 Discriminative coefficient term $f(\boldsymbol{X})$

To make the dictionary $\boldsymbol{D}$ be discriminative for the samples in $\boldsymbol{A}$, we can make the representation coefficient of $\boldsymbol{A}$ over $\boldsymbol{D}$, i.e. $\boldsymbol{X}$, be discriminative. Based on some criterion such as the Fisher discrimination criterion [252], this can be achieved by minimizing the within-class scatter of $\boldsymbol{X}$, denoted by $\boldsymbol{S}_W(\boldsymbol{X})$, and maximizing the between-class scatter of

$\boldsymbol{X}$, denoted by $\boldsymbol{S}_B(\boldsymbol{X})$. $\boldsymbol{S}_W(\boldsymbol{X})$ and $\boldsymbol{S}_B(\boldsymbol{X})$ are defined as

$$\boldsymbol{S}_W(\boldsymbol{X}) = \sum_{i=1}^{K} \sum_{\boldsymbol{x}_k \in \boldsymbol{X}_i} (\boldsymbol{x}_k - \boldsymbol{m}_i)(\boldsymbol{x}_k - \boldsymbol{m}_i)^T, \text{ and } \boldsymbol{S}_B(\boldsymbol{X}) = \sum_{i=1}^{K} n_i (\boldsymbol{m}_i - \boldsymbol{m})(\boldsymbol{m}_i - \boldsymbol{m})^T,$$

where $\boldsymbol{m}_i$ and $\boldsymbol{m}$ are the mean vector of $\boldsymbol{X}_i$ and $\boldsymbol{X}$ respectively, and n_i is the number of samples in class $\boldsymbol{A}_i$.

Intuitively, we can define $f(\boldsymbol{X})$ as $tr(\boldsymbol{S}_W(\boldsymbol{X})) - tr(\boldsymbol{S}_B(\boldsymbol{X}))$. However, the term of $-tr(\boldsymbol{S}_B(\boldsymbol{X}))$ makes such an $f(\boldsymbol{X})$ non-convex and unstable. To solve this problem, we propose to add an elastic term $\|\boldsymbol{X}\|_F^2$ into $f(\boldsymbol{X})$. So $f(\boldsymbol{X})$ is defined as

$$f(\boldsymbol{X}) = tr(\boldsymbol{S}_W(\boldsymbol{X})) - tr(\boldsymbol{S}_B(\boldsymbol{X})) + \eta\|\boldsymbol{X}\|_F^2, \tag{6-5}$$

where η is a parameter. We will further discuss the convexity of $f(\boldsymbol{X})$ in Section 6.4.

6.3.3 The FDDL model

By incorporating Eqs. (6-4) and (6-5) into Eq. (6-3), we have the following FDDL model:

$$\min_{\boldsymbol{D},\boldsymbol{X}} \left\{ \sum_{i=1}^{K} r(\boldsymbol{A}_i, \boldsymbol{D}, \boldsymbol{X}_i) + \lambda_1 \|\boldsymbol{X}\|_1 + \lambda_2 \left(tr(\boldsymbol{S}_W(\boldsymbol{X}) - \boldsymbol{S}_B(\boldsymbol{X})) + \eta \|\boldsymbol{X}\|_F^2 \right) \right\} \tag{6-6}$$

The term $\|\boldsymbol{X}\|_F^2$ not only makes the fidelity coefficient term convex but also makes the solving of $\boldsymbol{X}$ in Eq. (6-6) smoother due to that $\|\boldsymbol{X}\|_1$ is convex but not differentiable.

Although the objective function J in Eq. (6-6) is not jointly convex to ($\boldsymbol{D}$, $\boldsymbol{X}$), we will see that it is convex with respect to each of $\boldsymbol{D}$ and $\boldsymbol{X}$ when the other is fixed. Therefore, an algorithm of alternatively optimizing $\boldsymbol{D}$ and $\boldsymbol{X}$ can be designed. Detailed discussions and the optimization procedures are presented next in Section 6.4.

6.3.4 A simplified version of FDDL

The unconstraint problem of FDDL (Eq. (6-6)) could be rewritten as a constrained formulation:

$$\min_{\boldsymbol{D},\boldsymbol{X}}\left\{\sum_{i=1}^{K}\left(\left\|\boldsymbol{A}_i-\boldsymbol{D}\boldsymbol{X}_i\right\|_F^2+\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2\right)+\lambda_1\left\|\boldsymbol{X}\right\|_1+\lambda_2\left(tr\left(\boldsymbol{S}_W\left(\boldsymbol{X}\right)-\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right)+\eta\left\|\boldsymbol{X}\right\|_F^2\right)\right\} \quad \text{s.t. } \left\|\boldsymbol{D}_j\boldsymbol{X}_i^j\right\|_F^2\le\varepsilon_f,\ \forall i\neq j \tag{6-7}$$

where ε_f is a scalar. The constraint could guarantee that each class-specific sub-dictionary has poor representation ability for other classes.

It is a little complex to solve the FDDL model (i.e., Eq. (6-6) or Eq. (6-7)). One simplified way to solve the FDDL model is to find first the formulation which approximately meets the constraint in Eq. (6-7), and then to find the optimal one which follows that formulation. Inspired by the prior that $\boldsymbol{X}_i^j$ should have nearly zero representation on the dictionary $\boldsymbol{D}_j$, $j\neq i$, the simplified FDDL could be the one by assuming $\boldsymbol{X}_i^j=\boldsymbol{0}$ for $j\neq i$. In this case, the constraint can be well met since $\left\|\boldsymbol{D}_j\boldsymbol{X}_i^j\right\|_F^2=0$ for $j\neq i$. In the simplified FDDL, the sparse coding coefficient $\boldsymbol{X}$ becomes a block diagonal matrix, whose between-class scatter, $tr(\boldsymbol{S}_B(\boldsymbol{X}))$, could be shown to be large enough in general (please refer to **Appendix 1** for the details).

Based on the above discussions, the simplified FDDL model could be written as

$$\min_{\boldsymbol{D},\boldsymbol{X}}\left\{\sum_{i=1}^{K}\left(\left\|\boldsymbol{A}_i-\boldsymbol{D}\boldsymbol{X}_i\right\|_F^2+\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2\right)+\lambda_1\left\|\boldsymbol{X}\right\|_1+\lambda_2\left(tr\left(\boldsymbol{S}_W\left(\boldsymbol{X}\right)-\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right)+\eta\left\|\boldsymbol{X}\right\|_F^2\right)\right\} \quad \text{s.t. } \boldsymbol{X}_i^j=\boldsymbol{0},\ \forall i\neq j \tag{6-8}$$

which could be further formulated as (please refer to **Appendix 2** for the detailed derivation)

$$\min_{\boldsymbol{D},\boldsymbol{X}}\sum_{i=1}^{K}\left(\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2+\lambda_1'\left\|\boldsymbol{X}_i^i\right\|_1+\lambda_2'\left\|\boldsymbol{X}_i^i-\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|_F^2+\lambda_3'\left\|\boldsymbol{X}_i^i\right\|_F^2\right) \tag{6-9}$$

where $\lambda_1'=\lambda_1/2$, $\lambda_2'=\lambda_2\left(1+\kappa_i\right)/2$, κ_i=1-n_i/n, and $\lambda_3'=\lambda_2\left(\eta-\kappa_i\right)/2$. Here $\boldsymbol{m}_i^i$ and $\boldsymbol{m}_i$ are the mean vectors of $\boldsymbol{X}_i^i$ and $\boldsymbol{X}_i$, respectively. It can be seen that the dictionary learning in the simplified FDDL model could be performed class by class.

6.4 Optimization of FDDL

In this section, we first present the optimization procedure of original FDDL model in Eq. (6-6), and then present the solution of simplified FDDL model in Eq. (6-9).

The FDDL objective function in Eq. (6-6) can be divided into two sub-problems by optimizing $\boldsymbol{D}$ and $\boldsymbol{X}$ alternatively: updating $\boldsymbol{X}$ by fixing $\boldsymbol{D}$, and updating $\boldsymbol{D}$ by fixing $\boldsymbol{X}$. The procedures are iteratively implemented for the desired discriminative dictionary $\boldsymbol{D}$ and the discriminative coefficients $\boldsymbol{X}$.

6.4.1 Sparse coding of FDDL

First, suppose that the dictionary $\boldsymbol{D}$ is fixed, and the objective function $J_{(\boldsymbol{D},\boldsymbol{X})}$ in Eq. (6-6) is reduced to a sparse coding problem to compute $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K]$. Here we compute $\boldsymbol{X}_i$ class by class. When compute $\boldsymbol{X}_i$, all $\boldsymbol{X}_j$, $j \neq i$, are fixed. Thus the objective function in Eq. (6-6) is further reduced to:

$$\min_{\boldsymbol{X}_i} \left\{ r(\boldsymbol{A}_i, \boldsymbol{D}, \boldsymbol{X}_i) + \lambda_1 \left\| \boldsymbol{X}_i \right\|_1 + \lambda_2 f_i(\boldsymbol{X}_i) \right\} \tag{6-10}$$

with

$$f_i(\boldsymbol{X}_i) = \left\| \boldsymbol{X}_i - \boldsymbol{M}_i \right\|_F^2 - \sum_{k=1}^{K} \left\| \boldsymbol{M}_k - \boldsymbol{M} \right\|_F^2 + \eta \left\| \boldsymbol{X}_i \right\|_F^2,$$

where $\boldsymbol{M}_k$ and $\boldsymbol{M}$ are the mean vector matrices (by taking n_k mean vectors $\boldsymbol{m}_k$ or $\boldsymbol{m}$ as its column vectors) of class k and all classes, respectively. It can be proved that if $\eta > 1 - n_i/n$, $f_i(\boldsymbol{X}_i)$ is strictly convex to $\boldsymbol{X}_i$ (please refer to **Appendix 3** for the proof), where n_i and n are the number of training samples in the i^{th} class and all classes, respectively. In order to make $f_i(\boldsymbol{X}_i)$ not only convex but also have enough discrimination, in this thesis, we set $\eta = 1$ for simplicity. Then we can see that all the terms in Eq. (6-10), except for $\|\boldsymbol{X}\|_1$, are differentiable, and Eq. (6-10) is strictly convex. We rewrite Eq. (6-10) as

$$\min_{\boldsymbol{X}_i} \left\{ Q(\boldsymbol{X}_i) + 2\tau \left\| \boldsymbol{X}_i \right\|_1 \right\} \tag{6-11}$$

where $Q(\boldsymbol{X}_i)=r(\boldsymbol{A}_i,\boldsymbol{D},\boldsymbol{X}_i)+\lambda_2 f_i(\boldsymbol{X}_i)$, and $\tau=\lambda_1/2$. Define $\tilde{\boldsymbol{X}}_i=\left[\boldsymbol{x}_{i,1}^T,\boldsymbol{x}_{i,2}^T,\cdots,\boldsymbol{x}_{i,n_i}^T\right]^T$, where $\boldsymbol{x}_{i,k}$ is the k^{th} column vector of matrix $\boldsymbol{X}_i$. Because $Q(\boldsymbol{X}_i)$ is strictly convex and differentiable to $\boldsymbol{X}_i$, the Iterative Projection Method (IPM) [253] (whose speed could be improved by approaches like FISTA [201]) can be employed to solve Eq. (6-11), as described in Table 6.1.

Table 6.1: The sparse coding algorithm of FDDL.

Coding algorithm of FDDL

1. **Input**: σ, $\tau>0$.
2. **Initialization**: $\tilde{\boldsymbol{X}}_i^{(1)}=\boldsymbol{0}$ and h=1.
3. **While** convergence and the maximal iteration number are not reached **do**

   $h = h+1$

   $$\tilde{\boldsymbol{X}}_i^{(h)}=\boldsymbol{S}_{\tau/\sigma}\left(\tilde{\boldsymbol{X}}_i^{(h-1)}-\frac{1}{2\sigma}\nabla Q\left(\tilde{\boldsymbol{X}}_i^{(h-1)}\right)\right) \quad (6\text{-}12)$$

   where $\nabla Q\left(\tilde{\boldsymbol{X}}_i^{(h-1)}\right)$ is the derivative of $Q(\boldsymbol{X}_i)$ w.r.t. $\tilde{\boldsymbol{X}}_i^{(h-1)}$, and $\boldsymbol{S}_{\tau/\sigma}$ is a soft thresholding operator defined in component-wise [253] by:

   $$\left[\boldsymbol{S}_{\tau/\sigma}(\boldsymbol{x})\right]_j=\begin{cases}0 & |x_j|\le\tau/\sigma\\ x_j-\text{sign}(x_j)\tau/\sigma & \text{otherwise}\end{cases}.$$

4. **Return** $\tilde{\boldsymbol{X}}_i=\tilde{\boldsymbol{X}}_i^{(h)}$.

The sparse coding phase of simplified FDDL (i.e., Eq. (6-9)) is the special case with $Q(\boldsymbol{X}_i)=\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2+\lambda_2'\left\|\boldsymbol{X}_i^i-\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|_2^2+\lambda_3'\left\|\boldsymbol{X}_i^i\right\|_F^2$ and $\boldsymbol{X}_i^j=0$ for $j\neq i$, which could also be efficiently solved by the algorithm in Table 6.1. For simplified FDDL, we set $\eta=\kappa_i=1-n_i/n$ (i.e., $\lambda_3'=0$) in this thesis.

6.4.2 Dictionary updating of FDDL

Let's then discuss when $\boldsymbol{X}$ is fixed, how to update $\boldsymbol{D}=[\boldsymbol{D}_1,\boldsymbol{D}_2,\ldots,\boldsymbol{D}_K]$. In order to avoid

that $\boldsymbol{D}$ has arbitrarily large l_2-norm, which would result in trivial values of the coding coefficients, i.e., arbitrarily small values, we constrain each column vector of $\boldsymbol{D}$ to have a unit l_2-norm. We also update $\boldsymbol{D}_i = [\boldsymbol{d}_1, \boldsymbol{d}_2, ..., \boldsymbol{d}_{p_i}]$ class by class. When update $\boldsymbol{D}_i$, all $\boldsymbol{D}_j$, $j \neq i$, are fixed. Now the objective function in Eq. (6-6) is reduced to:

$$\min_{\boldsymbol{D}_i} \left\{ \left\| \hat{\boldsymbol{A}} - \boldsymbol{D}_i \boldsymbol{X}^i \right\|_F^2 + \left\| \boldsymbol{A}_i - \boldsymbol{D}_i \boldsymbol{X}_i^i \right\|_F^2 + \sum_{j=1, j\neq i}^{K} \left\| \boldsymbol{D}_i \boldsymbol{X}_j^i \right\|_F^2 \right\} \quad \text{s.t.} \left\| \boldsymbol{d}_l \right\|_2 = 1, l = 1, \cdots, p_i \tag{6-13}$$

where $\hat{\boldsymbol{A}} = \boldsymbol{A} - \sum_{j=1, j\neq i}^{K} \boldsymbol{D}_j \boldsymbol{X}^j$ and $\boldsymbol{X}^i$ is the coding coefficients of $\boldsymbol{A}$ over $\boldsymbol{D}_i$. Eq. (6-13) could be further re-written as

$$\min_{\boldsymbol{D}_i} \left\| \boldsymbol{\Lambda}_i - \boldsymbol{D}_i \boldsymbol{Z}_i \right\|_F^2 \text{ s.t.} \left\| \boldsymbol{d}_l \right\|_2 = 1, l = 1, \cdots, p_i \tag{6-14}$$

where $\boldsymbol{\Lambda}_i = \left[\hat{\boldsymbol{A}} \boldsymbol{A}_i \ \boldsymbol{0} \cdots \boldsymbol{0} \ \boldsymbol{0} \cdots \boldsymbol{0} \right]$ and $\boldsymbol{Z}_i = \left[\boldsymbol{X}^i \ \boldsymbol{X}_i^i \ \boldsymbol{X}_1^i \cdots \boldsymbol{X}_{i-1}^i \ \boldsymbol{X}_{i+1}^i \cdots \boldsymbol{X}_K^i \right]$. Eq. (6-14) is a quadratic programming problem, and its solution is the same as the dictionary updating parts of the algorithm of Gabor occlusion dictionary (GOD) computing in Table 4.2.

The dictionary updating of simplified FDDL is also the same as that of the original FDDL except that Eq. (6-14) becomes a simpler one with $\boldsymbol{\Lambda}_i = \boldsymbol{A}_i$ and $\boldsymbol{Z}_i = \boldsymbol{X}_i^i$.

6.4.3 Algorithm of FDDL

The whole algorithm of FDDL is summarized in Table 6.2. The algorithm converges since the two alternative optimizations (i.e., sparse coding of FDDL and dictionary updating of FDDL) in it are both convex.

Table 6.2: Algorithm of Fisher Discrimination Dictionary Learning.

Fisher Discrimination Dictionary Learning (FDDL)
1. Initialization *D*. We initialize all the p_i atoms of each $\boldsymbol{D}_i$ as random vectors with unit l_2-norm.
2. Sparse coding coefficients *X*. Fix $\boldsymbol{D}$ and solve $\boldsymbol{X}_i$, i=1,2,…,K, one by one by solving Eq. (6-11) with the algorithm in **Table 6.1**.
3. Updating dictionary *D*.

Fix $\boldsymbol{X}$ and update each $\boldsymbol{D}_i$, i=1,2,…,K, by solving Eq. (6-14) :

1) Rewrite $\boldsymbol{Z}_i = \left[\boldsymbol{z}_1;\boldsymbol{z}_2;\cdots;\boldsymbol{z}_{p_i}\right]$, $\boldsymbol{D}_i = \left[\boldsymbol{d}_1;\boldsymbol{d}_2;\cdots;\boldsymbol{d}_{p_i}\right]$ where $\boldsymbol{z}_j$, j=1,2,…,p_i, is the row vector of $\boldsymbol{Z}_i$, and $\boldsymbol{d}_j$ is the j^{th} column vector of $\boldsymbol{D}_i$.

2) Fix all $\boldsymbol{d}_l, l \neq j$, update $\boldsymbol{d}_j$. Let $\boldsymbol{Y} = \boldsymbol{A}_i - \sum_{l \neq j} \boldsymbol{d}_l \boldsymbol{\beta}_l$, so the optimization problem of minimization of Eq. (6-14) changes to

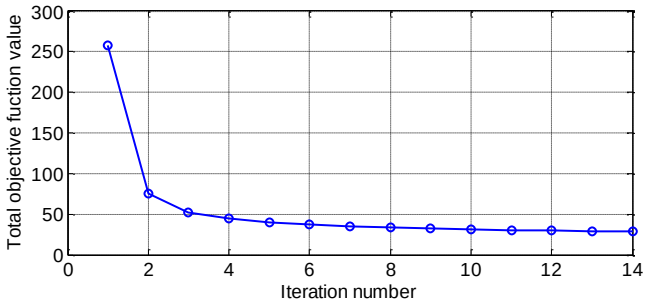
$$\min_{\boldsymbol{d}_j} \left\| \boldsymbol{Y} - \boldsymbol{d}_j \boldsymbol{\beta}_j \right\|_F^2 \quad \text{s.t.} \quad \left\| \boldsymbol{d}_j \right\|_2 = 1;$$

After some deviation, finally we could get the solution which is $\boldsymbol{d}_j = \boldsymbol{Y}\boldsymbol{\beta}_j^T \Big/ \left\| \boldsymbol{Y}\boldsymbol{\beta}_j^T \right\|_2$

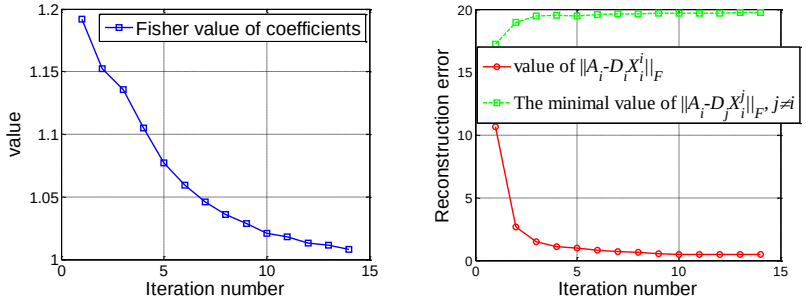
3) Using the above procedures, we can update all the vectors $\boldsymbol{d}_j$, and hence the whole dictionary $\boldsymbol{D}_i$ is updated.

4. Output.

Return to **step 2** until the object function values in adjacent iterations are close enough, or the maximum number of iterations is reached. Then output $\boldsymbol{X}$ and $\boldsymbol{D}$.



(a)



(b) (c)

Figure 6.2: An example of FDDL process on the Extended Yale B face database. (a) The convergence of FDDL. (b) The curve of $tr(\boldsymbol{S}_W(\boldsymbol{X}))/tr(\boldsymbol{S}_B(\boldsymbol{X}))$ versus iteration number. (c) The curves of the reconstruction error of $\boldsymbol{D}_i$ to $\boldsymbol{A}_i$ and the minimal reconstruction error of $\boldsymbol{D}_j$ to $\boldsymbol{A}_i$, $j \neq i$, versus the iteration number.

Fig. 6.2 shows an example of FDDL on the Extended Yale B face database. Fig. 6.2(a) illustrates the convergence of FDDL; Fig. 6.2(b) shows that the value of $tr(\boldsymbol{S}_W(\boldsymbol{X}))/tr(\boldsymbol{S}_B(\boldsymbol{X}))$ (essentially equivalent to $tr(\boldsymbol{S}_W(\boldsymbol{X}))-tr(\boldsymbol{S}_B(\boldsymbol{X}))$ in representing the discrimination ability of $\boldsymbol{X}$ but is invariant to the scale of $\boldsymbol{X}$) decreases as the iteration number increases, which indicates that the coefficients $\boldsymbol{X}$ are discriminative after learning the dictionary $\boldsymbol{D}$; Fig. 6.2(c) plots the curves of $\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\|_F$ (i=10 here) and the minimal value of $\|\boldsymbol{A}_i-\boldsymbol{D}_j\boldsymbol{X}_i^j\|_F$, j=1,2,…,K, $j\neq i$, showing that the dictionary $\boldsymbol{D}_i$ could represent the samples of $\boldsymbol{A}_i$ well, but $\boldsymbol{D}_j$, $j\neq i$, has poor representation ability to the samples of $\boldsymbol{A}_i$.

With FDDL, we could use the sparse coding coefficients of each class, i.e., $\boldsymbol{X}_i$, to compute the mean coefficient vector of that class, denoted by $\boldsymbol{m}_i$, which will then be used for the testing sample classification. For simplified FDDL, the mean coefficient vector for each class is constructed by $\boldsymbol{m}_i=\left[\boldsymbol{0};\cdots;\boldsymbol{m}_i^i;\cdots;\boldsymbol{0}\right]$, where $\boldsymbol{m}_i^i$ is the mean vector of $\boldsymbol{X}_i^i$.

6.5 The Classification Scheme

If the dictionary $\boldsymbol{D}$ is available, a testing sample can be classified via sparsely coding it over $\boldsymbol{D}$. Based on the employed dictionary $\boldsymbol{D}$, different information can be utilized for the classification task. In the methods [155, 157, 184, 196], a common dictionary is shared by all classes, and the sparse coding coefficients are used for classification. In the SRC scheme [102], the original training samples are used to form a structured dictionary to code the testing sample, and then the reconstruction error associated with each class is used for classification. Compared to SRC, in [158, 162] the testing sample is sparsely coded on each sub-dictionary associated with each class, and then the reconstruction error is computed for classification.

Although the methods in [102, 155, 157, 158, 162, 184, 196] could lead to good results, they are not able to use both the reconstruction errors and the coding coefficients

for image classification. With the proposed FDDL model in Eq. (6-6), however, the learnt dictionary $\boldsymbol{D}$ will make both the reconstruction error and the sparse coding coefficients discriminative. Naturally, we can make use of both the reconstruction error associated with each class and the coding coefficients for more accurate classification results.

According to the number of training samples per class, we propose two classification schemes, the global classifier (GC) and local classifier (LC), which use both the reconstruction error and the coding coefficients.

1) GC: When the number of training samples of each class is relatively small, the learnt dictionary $\boldsymbol{D}_i$ may not be able to faithfully represent the testing samples of this class, and hence we code the testing sample $\boldsymbol{y}$ over the whole dictionary $\boldsymbol{D}$. In this case, the sparse coding coefficients could be got by solving

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 + \gamma \|\boldsymbol{\alpha}\|_1 \right\} \tag{6-15}$$

where γ is a constant. Denote by $\hat{\boldsymbol{\alpha}} = [\hat{\boldsymbol{\alpha}}_1; \hat{\boldsymbol{\alpha}}_2; \cdots; \hat{\boldsymbol{\alpha}}_K]$, where $\hat{\boldsymbol{\alpha}}_i$ is the coefficient vector associated with sub-dictionary $\boldsymbol{D}_i$. We define the metric for final classification as

$$e_i = \|\boldsymbol{y} - \boldsymbol{D}_i \hat{\boldsymbol{\alpha}}_i\|_2^2 + w \cdot \|\hat{\boldsymbol{\alpha}} - \boldsymbol{m}_i\|_2^2 \tag{6-16}$$

where the first term is the reconstruction error by class i, the second term is the distance between the coefficient vector $\hat{\boldsymbol{\alpha}}$ and the learnt mean vector $\boldsymbol{m}_i$ of class i, and w is a preset weight to balance the contribution of the two terms. The classification of $\boldsymbol{y}$ is made by $\text{identity}(\boldsymbol{y}) = \arg\min_i \{e_i\}$.

2) LC: When the number of training samples of each class is relatively large, the learnt dictionary $\boldsymbol{D}_i$ is able to well span the sample space of class i, and thus we could directly code the testing sample $\boldsymbol{y}$ by $\boldsymbol{D}_i$ to reduce the computational cost and the interference of other dictionaries. The coding coefficients associated with $\boldsymbol{D}_i$ are got by solving

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\{ \|\boldsymbol{y} - \boldsymbol{D}_i\boldsymbol{\alpha}\|_2^2 + \gamma_1 \|\boldsymbol{\alpha}\|_1 + \gamma_2 \left\|\boldsymbol{\alpha} - \boldsymbol{m}_i^i\right\|_2^2 \right\} \tag{6-17}$$

where γ_1 and γ_2 are constants. Here we require not only that the sub-dictionary $\boldsymbol{D}_i$ should

well code the testing sample $\boldsymbol{y}$ with sparse coefficients, but also that the coding vector $\boldsymbol{\alpha}$ should be close to $\boldsymbol{m}_i^i$, the i^{th}-class trained mean vector associated with sub-dictionary $\boldsymbol{D}_i$. Hence the metric for final classification is defined as

$$e_i = \left\| \boldsymbol{y} - \boldsymbol{D}_i \hat{\boldsymbol{\alpha}} \right\|_2^2 + \gamma_1 \left\| \hat{\boldsymbol{\alpha}} \right\|_1 + \gamma_2 \left\| \hat{\boldsymbol{\alpha}} - \boldsymbol{m}_i^i \right\|_2^2 \tag{6-18}$$

The final classification rule is also $\text{identity}(\boldsymbol{y}) = \arg\min_i \{e_i\}$.

6.6 Experimental Results

In this section, we verify the performance of FDDL on applications such as FR, digit recognition, gender classification and object categorization. The selection of model and parameters is discussed in Section 6.6.1. Then face recognition, digit recognition, gender classification and object categorization are performed by using FDDL and the competing methods in Section 6.6.2, Section 6.6.3, Section 6.6.4 and Section 6.6.5, respectively.

6.6.1 Model and parameter selection

In this section, the selection of dictionary learning model (e.g., FDDL and simplified FDDL), classification model (e.g., GC and LC), the number of dictionary atoms, l_1-norm or l_2-norm regularization, and parameters is discussed. In order to better analyze the model selection, the parameters in classifiers, i.e., γ and w in GC and γ_1 and γ_2 in LC, are predefined. Specifically, we set the values of γ and γ_1 from set {0.001, 0.005, 0.01, 0.05, 0.1}, and set the values of w and γ_2 from set {0, 0.001}. Given a dictionary training and classification model, we report the best performance of different classifiers.

6.6.1.1 Model selection in dictionary learning and classification

Here we evaluate how to select dictionary learning and classification model. As mentioned before, simplified FDDL is a special but important case of FDDL model by assuming $\boldsymbol{X}_i^j = \boldsymbol{0}$ for $j \neq i$. One advantage of simplified FDDL lies in that the dictionary

learning for each class could be performed in parallel. When there are (nearly) enough training samples for each class, it is not necessary to represent the samples on the whole dictionary of all classes. Therefore the simplification of $\boldsymbol{X}_i^j = \mathbf{0}$ for $j \neq i$ is very reasonable. In the phase of classification, the GC and LC classifiers are suitable for small-sample-size problem and enough-training-sample problem, respectively.

The FR performance of FDDL or simplified FDDL coupled with GC or LC on the AR database (the detailed experimental settings can be found in Section 6.6.2) [212] is listed in Table 6.3. Here we set λ_1=0.005 and λ_2=0.01 in dictionary learning. It can be seen that with FDDL or simplified FDDL the GC achieves much better performance than the LC with over 20% gap under different numbers of training samples. In addition, FDDL and its simplified version have similar performance.

Table 6.3: FR rates of FDDL and simplified FDDL coupled with GC or LC on the AR database.

Trainning num	4		7	
	GC	LC	GC	LC
FDDL	***0.863***	*0.615*	0.926	*0.748*
Simplified FDDL	0.860	0.614	***0.930***	*0.748*

Table 6.4: Performance of FDDL and simplified FDDL coupled with GC or LC in USPS digit recognition.

Training num	5		10		100		300	
	GC	LC	GC	LC	GC	LC	GC	LC
FDDL	*0.789*	***0.798***	*0.829*	***0.843***	0.902	0.941	0.908	0.941
Simplified FDDL	0.785	0.795	0.829	0.841	*0.929*	***0.942***	*0.943*	***0.950***

Table 6.4 compares the performance of FDDL and simplified FDDL coupled with GC and LC in the USPS [254] digit recognition (the detailed experimental settings can be found in Section 6.6.3). We set λ_1=0.05 and λ_2=0.005. Opposite to that of FR, with either FDDL or simplified FDDL the LC always outperforms the GC, especially when the number of training samples increases. The recognition rates of FDDL and simplified

FDDL with LC are very close. On the other hand, when the number of training samples is not enough (e.g., 5 and 10), FDDL gets a little higher recognition rate, while the simplified FDDL is slightly better in the case that the number of training samples is big.

In order to reduce the computational burden, we adopt simplified FDDL to learn the dictionary in all the experiments except for the part of face recognition. In the classification phase, GC is used in face recognition and object categorization, LC is used in digit recognition, and both GC and LC are tested in gender classification.

6.6.1.2 Discussion on the number of dictionary atoms

One important parameter in FDDL is the number of atoms in $\boldsymbol{D}_i$, denoted by p_i. For FDDL, we usually set all p_i equal, i=1,2,…,K. We use SRC as the baseline method, and analyze the effect of p_i on the performance of FDDL. We take FR on Extended Yale B [99, 206] as an example (the experimental setting is given in Section 6.6.2). Because SRC uses the original training samples as dictionary, we randomly select p_i training samples as dictionary atoms and run 10 times the experiment to get the average recognition rate. Fig. 6.3 plots the recognition rates of FDDL and SRC versus different number of dictionary atoms. We can see that in all cases FDDL has about 3% improvement over SRC. Especially, even with the atom number p_i=8, FDDL can still have higher recognition rate than SRC with p_i=20. Besides, from p_i=20 to p_i=8, FDDL's recognition rate drops by 2.2%, compared to 4.2% for SRC. This indicates that FDDL is effective to compute a compact and representative dictionary, which can reduce the computational cost and improve the recognition rate simultaneously.

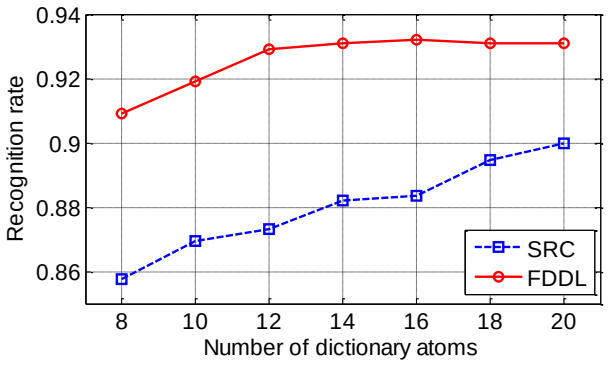


Figure 6.3: The recognition rates of FDDL and SRC versus the number of dictionary atoms.

6.6.1.3 Discussions on l_1-norm regularization and l_2-norm regularization

In Chapter 3 we have indicated that the sparse constraint on the coding coefficients may not be necessary in the coding stage of SRC. In the proposed FDDL model, both l_1-norm and l_2-norm regularizations are imposed on the coding coefficients $\boldsymbol{X}$ in the stage of learning the dictionary. In this sub-section, we evaluate the role of l_1-norm regularization term (i.e., $\|\boldsymbol{X}\|_1$ in Eq. (6.6)) and l_2-norm regularization term (i.e., $\|\boldsymbol{X}\|_F^2$ in Eq. (6-6)) in the dictionary learning phase by changing the values of parameters λ_1 and η (η>1-n_i/n). We also evaluate the performance of GC and LC with l_2-norm regularization in coding the testing sample (i.e., replacing $\|\boldsymbol{\alpha}\|_1$ by $\|\boldsymbol{\alpha}\|_F^2$ in Eq. (6-15) and Eq. (6-17)).

Table 6.5 lists the recognition rates of FDDL with different values of λ_1 and η on the Extended Yale B database (the experimental setting is given in Section 6.6.2). The GC is used in this experiment for classification. Here we set λ_2=0.005. It can be seen that when there is no l_1-norm regularization on the coding coefficient (i.e., λ_1 =0) in FDDL, the performance will degrade (e.g., the performance when λ_1 =0 is lower than that when λ_1 =0.005). When there is l_1-norm regularization (e.g., λ_1 =0.005), the varying strength of l_2-norm regularization has little affect on the final performance. This finding shows that l_1-norm regularization is useful and important in learning discriminative dictionary for pattern classification.

Table 6.5: FR rates on the Extended Yale B database with various parameter settings of (λ_1, η).

Parameters	(0.005, κ_i)	(0.005, 1)	(0.005, 5)	(0, 1)	(0, 5)
L_1-norm GC	**0.924**	0.921	0.924	0.908	0.917
L_2-norm GC	0.911	0.907	0.913	0.886	0.916

With the learnt dictionary by FDDL, in the classification stage, from Table 6.5 we can see that the l_2-norm regularized GC has a little bit lower recognition rates than the l_1-norm regularized GC. This is mainly because that l_1-norm regularization is used in learning the dictionary, so if l_1-norm regularization is not employed in coding the testing sample, the discrimination of the dictionary may not be fully exploited.

We then apply FDDL to the UPSP digit database with 300 training samples, with the recognition rates listed in Table 6.6. The LC classifier is used in this experiment. Here we also change the values of λ_1 and η, and fix λ_2=0.005. Similar conclusions could be made: l_1-norm sparse regularization is useful in the phase of dictionary learning and consequently the l_1-norm regularized classifier is more powerful than l_2-norm regularized classifier in couple with the learnt dictionary.

Table 6.6: Digit recognition rate on the USPS database with various parameter settings of (λ_1, η).

parameters	(0.05, κ_i)	(0.05, 1)	(0.05, 5)	(0, 1)	(0, 5)
L_1-norm LC	0.950	0.950	**0.952**	0.931	0.933
L_2-norm LC	0.933	0.933	0.934	0.916	0.933

6.6.1.4 Cross-validation of parameters

In all the experiments, if no specific instructions, the tuning parameters in FDDL (λ_1 and λ_2 in dictionary learning phase, γ and w in GC or γ_1 and γ_2 in LC) and the parameters of competing methods are evaluated by 5-fold cross validation to avoid over-fitting. Because there are many combinations of these four parameters, we use a few simple heuristic to

reduce the search space. Usually, the value range of λ_1 and λ_2 is between 0.001 and 0.1 and the value of γ or γ_1 is similar to the value of λ_1. Therefore, w or γ_2 is set as 0 to firstly search the optimal values of λ_1, λ_2 and γ or γ_1; then with the fixed λ_1, λ_2 and γ or γ_1, the value of w or γ_2 is searched. We usually choose the set {0.001, 0.005, 0.01, 0.05, and 0.1} as the set of optimal values of λ_1, λ_2 and γ or γ_1.

6.6.2 Face recognition

In this section, we apply the proposed algorithm to FR on the Extended Yale B [99, 206], AR [212], and Multi-PIE [213] face databases. In order to clearly illustrate the advantage of the proposed method, besides SRC we compare FDDL with two latest sparse-dictionary-learning based classification methods, *discriminative KSVD* (DKSVD) [157] and *dictionary learning with structure incoherence* (DLSI) [158], and two popular classification methods, *nearest neighbor* (NN) and linear *support vector machines* (SVM). Note that the original DLSI method codes the testing sample by each class. For a fair comparison, we also gave the results (denoted by DLSI*) by coding the testing sample on the whole dictionary and using the reconstruction error for classification. The default number of dictionary atoms in FDDL on each class is set as the number of training samples. The Eigenface [57] with dimension 300 is used in all FR experiments.

a) FR on Extended Yale B database: The Extended Yale B database consists of 2,414 frontal-face images from 38 individuals (about 64 images per subject), captured under various laboratory-controlled lighting conditions. For each subject, we randomly selected 20 images for training, with the others (about 44 images per subject) for testing. The images were normalized to 54×48. The results of FDDL, SRC, NN, SVM, DKSVD and DLSI are listed in Table 6.7. It can be seen that FDDL can improve about at least 2% over all the other methods. DKSVD, which only uses sparse coefficients to do classification, does not work well here. DLSI* has better results than DLSI, which shows that coding the testing image on the whole dictionary is more reasonable in this case.

Table 6.7: The FR rates of various methods on the Extended Yale B database.

Method	SRC	NN	SVM	DKSVD	DLSI (DLSI*)	**FDDL**
Reco-rate	0.900	0.617	0.888	0.753	0.850 (0.890*)	**0.919**

b) FR on the AR database: The AR database consists of over 4,000 frontal images from 126 individuals. For each individual, 26 pictures were taken in two separated sessions. As in [102], in the experiment we chose a subset consisting of 50 male subjects and 50 female subjects. For each subject, the 7 images with illumination and expression changes from Session 1 were used for training, and the other 7 images with the same condition from Session 2 were used for testing. The size of original face image is 60×43. The comparison of FDDL with the competing methods is shown in Table 6.8. Again, we can see that FDDL has at least 3% improvement over the other methods. DLSI* has the second best performance; however, DLSI gets the second worst results because each class has only 7 training samples in this experiment.

Table 6.8: The FR rates of various methods on the AR database.

Method	SRC	NN	SVM	DKSVD	DLSI(DLSI*)	**FDDL**
Reco-rate	0.888	0.714	0.871	0.854	0.737 (0.898*)	**0.920**

c) FR on the Multi-PIE database: The CMU Multi-PIE face database [213] is a large scale database of 337 subjects including four sessions with simultaneous variations of pose, expression and illumination. Among the 337 subjects, we chose the first 60 subjects presented in Session 1 as the training set to do FR. For each of the 60 training subjects, we used the frontal images of 14 illuminations[5], taken with neutral expression (for Test 1) or smile expression (for Test 2), for training. For the testing set, we used the frontal images of 10 illuminations[6] from Session 3 with neutral expression (for Test 1) or smile expression (for Test 2). Note that Session 1 and Session 3 were recorded with long time

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

interval. The images were manually cropped and normalized to 100×82.

For FDDL, the dictionary size of each class is set as half of the number of training samples. The experimental results of different methods are listed in Table 6.9. We can see that compared with the previous methods, FDDL has at least 1% (in Test 1) or 2% (in Test 2) improvement with a smaller dictionary. SRC works the second best. DLSI* advances DLSI in all tests.

In all the FR experiments, DLSI* advances DLSI, and DKSVD is worse than FDDL, SRC and DLSI*, which may imply that the reconstruction error associated with each class is more powerful than the coding coefficients in face classification.

Table 6.9: The FR rates of various methods on the Multi-PIE database.

Method	SRC	NN	SVM	DKSVD	DLSI (DLSI*)	**FDDL**
Test 1	0.955	0.902	0.916	0.939	0.914 (0.941*)	**0.967**
Test 2	0.961	0.947	0.922	0.898	0.949 (0.959*)	**0.980**

6.6.3 Digit recognition

We then perform handwritten digit recognition on the widely used USPS database [254] with 7,291 training and 2,007 testing images. We compare the proposed FDDL with state-of-the-art methods reported in [144, 156, 158]. These methods include the best reconstructive DL method with linear and bilinear classifier models (denoted by REC-L and REC-BL) [156], the best supervised DL method with generative training and discriminative training (denoted by SDL-G and SDL-D) [156], the best result of sparse representation for signal classification (denoted by SRSC) [144] and the best result of DLSI [158]. In addition, some results of problem-specific methods (i.e., the standard Euclidean k_NN and SVM with a Gaussian kernel) reported in [158] are also listed. Here the dictionary of each class has 90 atoms in FDDL with $\lambda_1 = \gamma_1 = 0.1$, $\lambda_2 = 0.001$, and $\gamma_2 = 0.005$.

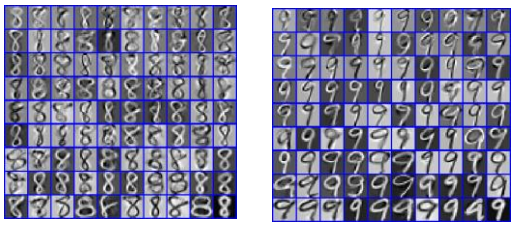


Figure 6.4: The learnt bases of digits 8 and 9 by FDDL.

Fig. 6.4 illustrates the learnt bases of digits 8 and 9. Table 6.10 lists the results of FDDL and its competing methods. We see that FDDL outperforms all the competing methods except for SDL-D (FDDL and SDL-D have very close results). It should be noted that SDL-D uses more information in DL and classification, including a learnt classifier of coefficients, the sparsity of coefficients, and the reconstruction error. In addition, the optimization of SDL-D method is much more complex than that of FDDL.

Table 6.10: Error rates of various methods on digit recognition.

Algorithms	**FDDL**	SRSC	REC-L	REC-BL	SDL-G	**SDL-D**	DLSI	KNN	SVM
Error rate (%)	**3.69**	6.05	6.83	4.38	6.67	**3.54**	3.98	5.2	4.2

6.6.4 Gender classification

In this experiment we chose a non-occluded subset (14 images per subject) of AR consisting of 50 male subjects and 50 female subjects. Images of the first 25 males and 25 females were used for training, and images of the remaining 25 males and 25 females for testing. We used PCA [57] to reduce the dimension of each image to 300. In addition, we present the result of DLSI$^{\#}$ (coding on the whole dictionary and classifying like SRC). Here p_i is set as 250 for FDDL and RBF kernel is adopted in SVM.

Table 6.11: The results of different methods on gender classification using the AR database.

SRC	DK-SVD	DLSI (DLSI#)	**LC (GC) with FDDL**	SVM	NN
0.930	0.861	0.940 (0.900)	**0.954 (0.941)**	0.924	0.907

Table 6.11 lists the recognition results of FDDL and the competing methods. It can be seen that LC with FDDL gets the best result when coding the testing image on the dictionary of each class (1.4% improvement compared to the second best one, DLSI); while GC with FDDL gets the best result when coding the testing sample on the whole dictionary (1.1% higher than the second best one, SRC). Meanwhile, we can see that DLSI and LC with FDDL have better performance than DLSI# and GC with FDDL respectively. This is because in gender recognition, there are only two classes and each class has enough training samples so that the learnt dictionary of each class is representative enough for the testing sample.

6.6.5 Object categorization

At last, let's validate the effectiveness of the proposed method on multi-class object categorization. An Oxford flower dataset with 17 categories [255] are used here, some samples of which are shown in Fig. 6.5. We adopt the default experimental settings provided on the website (www.robots.ox.ac.uk/˜vgg/data/flowers) including the training, validation, test splits and the multiple features. It should be noted that these features are only extracted from flower regions which are well cropped by the preprocessing of segmentation.

Figure 6.5: Samples of 'daffodil' from the Oxford flower data sets.

For a fair comparison with the state-of-the-art methods, e.g., MTJSRC [105], we also extended the original features from [255-256] to its kernel versions for the experiments on these two dataset. Specifically, we adopt the so-called column generation method [105]. Given the original training dataset $\boldsymbol{A}$ and a testing sample $\boldsymbol{y}$, the column-generation training dataset $\boldsymbol{G}$ and testing sample $\boldsymbol{h}$ are computed by $\boldsymbol{G}=\phi(\boldsymbol{A})^T\phi(\boldsymbol{A})$ and $\boldsymbol{h}=\phi(\boldsymbol{A})^T\phi(\boldsymbol{y})$, with $\phi(\boldsymbol{A})=[\ \phi(\boldsymbol{a}_1),\ldots,\ \phi(\boldsymbol{a}_n)]$, where $\boldsymbol{a}_i$ the i^{th} training sample and ϕ is the map function of the kernel defined by $\phi(\boldsymbol{a})^T\phi(\boldsymbol{y})=\exp\left(-\Omega(\boldsymbol{a},\boldsymbol{y})/\mu\right)$. Here μ is set to the mean value of the pairwise Chi-square distances (denoted by Ω) on the training set.

Table 6.12: The accuracy (mean±std %) performance by using single feature on the 17 category Oxford Flowers dataset.

Features	NS	SVM [257]	MTJSRC-CG [105]	SRC	FDDL+SRC
Color	61.7±3.3	60.9±2.1	64.0±3.3	61.9±2.2	**65.0±2.4**
Shape	69.9±3.2	70.3±1.3	71.5±0.8	72.7±1.9	**72.8±1.7**
Texture	55.8±1.4	63.7±2.7	**67.6±2.2**	61.4±0.9	64.9±1.7
HSV	61.3±0.7	62.9±2.3	65.0±3.9	62.5±3.0	**65.5±3.4**
HOG	57.4±3.0	58.5±4.5	62.6±2.7	61.4±1.9	**62.7±2.4**
SIFTint	70.7±0.7	70.6±1.6	74.0±2.2	73.7±2.9	**74.4±2.6**
SIFTbdy	61.9±4.2	59.4±3.3	63.2±3.3	62.3±2.6	**64.0±2.4**

Table 6.13: The accuracy (mean ± std %) performance by combining all features on the 17 category Oxford Flowers dataset.

Methods	Accuracy (%)
SRC combination	85.9±2.2
MKL [257]	85.2±1.5
LP-Boost [257]	85.4±2.4
CG-Boost [257]	84.8±2.2
MTJSRC-CG[105]	87.5±1.5
FDDL+SRC	86.7±1.3
FDDL+MTJSRC	**87.7±1.9**

This flower dataset contains 17 species of flowers with 80 images per class. As in [105], we directly use the χ^2 distance matrices of seven features (i.e., HSV, HOG, SIFTint, SIFTbdy, color, shape and texture vocabularies) to generate the column-generation training matrix $\boldsymbol{G}$ and testing samples $\boldsymbol{h}$. We firstly evaluate the performance of FDDL on each single feature. Table 6.12 lists the best results of NS, SVM, MTJSRC-CG, SRC and the proposed FDDL+SRC (i.e., w=0 in Eq. (6-16) for simplicity). It is clear to see that the dictionary learnt by FDDL could always improve the original SRC which directly uses training samples as the dictionary. Compared to the other state-of-the-art methods, such as SVM and MTJSRC, FDDL+SRC could achieve the highest recognition rates in most cases. We also evaluate the recognition performance by combing all features. In order to make a fair comparison of MTJSRC based on multi-task joint sparse representation, we give the results of FDDL+MTJSRC additionally. We simply set all the task weights in MTJSRC and FDDL+MTJSRC as 1. The results of FDDL (λ_1=0.005, λ_2=0.01) compared with other state-of-the-art methods are shown in Table 6.13. All MTJSRC, FDDL+SRC, and FDDL+MTJSC could achieve over 86.5% recognition rates, higher than all the other state-of-the-art methods. FDDL+SRC is slightly worse than MTJSRC, but FDDL+MTJSRC is slightly better than MTJSRC, which shows the effectiveness of FDDL in the same classification scheme.

6.7 Summary

In this Chapter, we proposed a Fisher Discrimination Dictionary Learning (FDDL) approach to sparsity based image classification. The FDDL model aims to learn a structured dictionary whose sub-dictionaries have specific class labels. The discrimination ability of FDDL is two-folds. First, each sub-dictionary of the learnt whole dictionary has good representation power to the samples from the corresponding class, but has poor representation power to the samples from other classes. Second, FDDL will result in discriminative coefficients by minimizing the with-class scatter and maximizing the between-class scatter of them. Consequently, we presented the classification schemes associated with FDDL, which use both the discriminative reconstruction error and sparse coding coefficients to classify the input testing image. The experimental results on face recognition, digit recognition, gender classification and object categorization clearly demonstrated the superiority of FDDL to many state-of-the-art dictionary learning based methods.

6.8 Appendix

Appendix 1: $tr(\boldsymbol{S}_B(\boldsymbol{X}))$ when $\boldsymbol{X}_i^j = 0,\ j \neq i$

Denote by $\boldsymbol{m}_i^i$, $\boldsymbol{m}_i$ and $\boldsymbol{m}$ the mean vectors of $\boldsymbol{X}_i^i$, $\boldsymbol{X}_i$ and $\boldsymbol{X}$, respectively. Because $\boldsymbol{X}_i^j = 0$ for $j \neq i$, we can rewrite $\boldsymbol{m}_i = \left[\mathbf{0};\cdots;\boldsymbol{m}_i^i;\cdots;\mathbf{0}\right]$ and $\boldsymbol{m} = \left[n_1\boldsymbol{m}_1^1;\cdots;n_i\boldsymbol{m}_i^i;\cdots;n_K\boldsymbol{m}_K^K\right]/n$. So the between-class scatter, i.e., $tr\left(\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right) = \sum_{i=1}^{K} n_i \left\|\boldsymbol{m}_i - \boldsymbol{m}\right\|_2^2$, changes to

$$\boldsymbol{S}_B\left(\boldsymbol{X}\right) = \sum_{i=1}^{K} n_i / n^2 \left[-n_1\boldsymbol{m}_1^1;\cdots;\left(n-n_i\right)\boldsymbol{m}_2^2;\cdots;-n_K\boldsymbol{m}_K^K\right]\left[-n_1\boldsymbol{m}_1^1;\cdots;\left(n-n_i\right)\boldsymbol{m}_2^2;\cdots;-n_K\boldsymbol{m}_K^K\right]^T.$$

Denote by κ_i=1-n_i/n, after some derivations the trace of $\boldsymbol{S}_B(\boldsymbol{X})$ becomes

$$tr\left(\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right)=\sum_{i=1}^{K}n_i/n^2\left\|\left[-n_1\boldsymbol{m}_1^1;\cdots;\left(n-n_i\right)\boldsymbol{m}_i^i;\cdots;-n_K\boldsymbol{m}_K^K\right]\right\|_2^2=\sum_{i=1}^{K}\kappa_i n_i\left\|\boldsymbol{m}_i^i\right\|_2^2 .$$

Because $\boldsymbol{m}_i^i$ is the mean coding vector of the samples from the same class, $\left\|\boldsymbol{m}_i^i\right\|_2^2$ often has a big energy value.

Appendix 2: The derivation of simplified FDDL model

Denote by $\boldsymbol{m}_i^i$ and $\boldsymbol{m}_i$ the mean vector of $\boldsymbol{X}_i^i$ and $\boldsymbol{X}_i$, respectively. Because $\boldsymbol{X}_i^j=0$ for $j \neq i$, we can rewrite $\boldsymbol{m}_i=\left[\boldsymbol{0};\cdots;\boldsymbol{m}_i^i;\cdots;\boldsymbol{0}\right]$. So the within-class scatter changes to

$$\boldsymbol{S}_W\left(\boldsymbol{X}\right)=\sum_{i=1}^{K}\sum_{\boldsymbol{x}_k\in\boldsymbol{X}_i}\left(\boldsymbol{x}_k^i-\boldsymbol{m}_i^i\right)\left(\boldsymbol{x}_i^i-\boldsymbol{m}_i^i\right)^T .$$

And the trace of within-class scatter is

$$tr\left(\boldsymbol{S}_W\left(\boldsymbol{X}\right)\right)=\sum_{i=1}^{K}\sum_{\boldsymbol{x}_k\in X_i}\left\|\boldsymbol{x}_k^i-\boldsymbol{m}_i^i\right\|_2^2 .$$

Based on **Appendix 1**, the trace of between-class scatter is $tr\left(\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right)=\sum_{i=1}^{K}\kappa_i n_i\left\|\boldsymbol{m}_i^i\right\|_2^2$, where κ_i=1-n_i/n. Therefore the discriminative coefficient term, i.e., $f\left(\boldsymbol{X}\right)=\left(\boldsymbol{S}_W\left(\boldsymbol{X}\right)-\boldsymbol{S}_B\left(\boldsymbol{X}\right)\right)+\eta\left\|\boldsymbol{X}\right\|_F^2$, could be simplified to

$$f\left(\boldsymbol{X}\right)=\sum_{i=1}^{K}\left(\sum_{\boldsymbol{x}_k\in X_i}\left\|\boldsymbol{x}_k^i-\boldsymbol{m}_i^i\right\|_2^2+\kappa_i\left(\left\|\boldsymbol{X}_i^i\right\|_F^2-n_i\left\|\boldsymbol{m}_i^i\right\|_2^2\right)+\left(\eta-\kappa_i\right)\left\|\boldsymbol{X}_i^i\right\|_F^2\right).$$

Denote by $\boldsymbol{E}_i^j=\left[1\right]_{n_i\times n_j}$ a matrix of size $n_i \times n_j$ with all entries being 1, then $\left[\boldsymbol{m}_i^i\right]_{1\times n_i}=\boldsymbol{X}_i^i\boldsymbol{E}_i^i/n_i$. Because $\boldsymbol{I}-\boldsymbol{E}_i^i/n_i\left(\boldsymbol{E}_i^i/n_i\right)^T=(\boldsymbol{I}-\boldsymbol{E}_i^i/n_i)(\boldsymbol{I}-\boldsymbol{E}_i^i/n_i)^T$, we get

$$\left\|\boldsymbol{X}_i^i\right\|_F^2-n_i\left\|\boldsymbol{m}_i^i\right\|_2^2=\left\|\boldsymbol{X}_i^i\right\|_F^2-\left\|\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|_F^2=tr\left(\boldsymbol{X}_i^i\left(\boldsymbol{I}-\boldsymbol{E}_i^i/n_i\left(\boldsymbol{E}_i^i/n_i\right)^T\right)\left(\boldsymbol{X}_i^i\right)^T\right)=\left\|\boldsymbol{X}_i^i-\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|$$

Then the discriminative coefficient term could finally be written as

$$f\left(\boldsymbol{X}\right)=\sum_{i=1}^{K}\left(\left(\kappa_i+1\right)\left\|\boldsymbol{X}_i^i-\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|_F^2+\left(\eta-\kappa_i\right)\left\|\boldsymbol{X}_i^i\right\|_F^2\right) \qquad (6\text{-}19)$$

With the constraint that $\boldsymbol{X}_i^j=0$ for $j\neq i$ in Eq. (6-8), we get

$$\left\|\boldsymbol{A}_i-\boldsymbol{D}\boldsymbol{X}_i\right\|_F^2=\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2 \tag{6-20}$$

With Eq. (6-19) and Eq. (6-20), the model of simplified FDDL (i.e., Eq. (6-8)) could be written as

$$\min_{\boldsymbol{D},\boldsymbol{X}}\sum\nolimits_{i=1}^{K}\left(\left\|\boldsymbol{A}_i-\boldsymbol{D}_i\boldsymbol{X}_i^i\right\|_F^2+\lambda_1'\left\|\boldsymbol{X}_i^i\right\|_1+\lambda_2'\left\|\boldsymbol{X}_i^i-\left[\boldsymbol{m}_i^i\right]_{1\times n_i}\right\|_F^2+\lambda_3'\left\|\boldsymbol{X}_i^i\right\|_F^2\right) \tag{6-21}$$

where $\lambda_1'=\lambda_1/2$, $\lambda_2'=\lambda_2\left(1+\kappa_i\right)/2$, and $\lambda_3'=\lambda_2\left(\eta-\kappa_i\right)/2$.

Appendix 3: The convexity of $f_i(\boldsymbol{X})$

Let $\boldsymbol{E}_i^j=[1]_{n_i\times n_j}$ be a matrix of size $n_i \times n_j$ with all entries being 1, and let $\boldsymbol{N}_i=\boldsymbol{I}_{n_i\times n_i}-\boldsymbol{E}_i^i/n_i$, $\boldsymbol{P}_i=\boldsymbol{E}_i^i/n_i-\boldsymbol{E}_i^i/n$, $\boldsymbol{C}_i^j=\boldsymbol{E}_i^j/n$, where $\boldsymbol{I}_{n_i\times n_i}$ is an identity matrix with the size of $n_i \times n_i$.

From $f_i\left(\boldsymbol{X}_i\right)=\left\|\boldsymbol{X}_i-\boldsymbol{M}_i\right\|_F^2-\sum_{k=1}^{K}\left\|\boldsymbol{M}_k-\boldsymbol{M}\right\|_F^2+\eta\left\|\boldsymbol{X}_i\right\|_F^2$, we can derive that

$$f_i\left(\boldsymbol{X}_i\right)=\left\|\boldsymbol{X}_i\boldsymbol{N}_i\right\|_F^2-\left\|\boldsymbol{X}_i\boldsymbol{P}_i-\boldsymbol{G}\right\|_F^2-\sum_{k=1,k\neq i}^{K}\left\|\boldsymbol{Z}_k-\boldsymbol{X}_i\boldsymbol{C}_i^k\right\|_F^2+\eta\left\|\boldsymbol{X}_i\right\|_F^2 \tag{6-22}$$

where $\boldsymbol{G}=\sum_{k=1,k\neq i}^{K}\boldsymbol{X}_k\boldsymbol{C}_k^i$, $\boldsymbol{Z}_k=\boldsymbol{X}_k\boldsymbol{E}_k^k/n_k-\sum_{j=1,j\neq i}^{K}\boldsymbol{X}_j\boldsymbol{C}_j^k$.

Rewrite $\boldsymbol{X}_i$ as a column vector, $\boldsymbol{\chi}_i=\left[\boldsymbol{r}_{i,1},\boldsymbol{r}_{i,2},\cdots,\boldsymbol{r}_{i,d}\right]^T$, where $\boldsymbol{r}_{i,j}$ is the j^{th} row vector of $\boldsymbol{X}_i$, and d is the total number of row vectors in $\boldsymbol{X}_i$. Then $f_i(\boldsymbol{X}_i)$ equals to

$$\left\|\operatorname{diag}\left(\boldsymbol{N}_i^T\right)\boldsymbol{\chi}_i\right\|_2^2-\left\|\operatorname{diag}\left(\boldsymbol{P}_i^T\right)\boldsymbol{\chi}_i-\operatorname{diag}\left(\boldsymbol{G}^T\right)\right\|_2^2-\sum_{k=1,k\neq i}^{K}\left\|\operatorname{diag}\left(\left(\boldsymbol{C}_i^k\right)^T\right)\boldsymbol{\chi}_i-\operatorname{diag}\left(\boldsymbol{Z}_k^T\right)\right\|_2^2+\eta\left\|\boldsymbol{\chi}_i\right\|$$

where diag($\boldsymbol{T}$) is to construct a block diagonal matrix with each block on the diagonal being matrix $\boldsymbol{T}$.

The convexity of $f_i(\boldsymbol{\chi}_i)$ depends on whether its Hessian matrix $\nabla^2 f_i(\boldsymbol{\chi}_i)$ is positive definite or not [123]. We could write the Hessian matrix of $f_i(\boldsymbol{\chi}_i)$ as

$$\nabla^2 f_i\left(\boldsymbol{\chi}_i\right)=2\operatorname{diag}\left(\boldsymbol{N}_i\boldsymbol{N}_i^T\right)-2\operatorname{diag}\left(\boldsymbol{P}_i\boldsymbol{P}_i^T\right)-\sum_{k=1,k\neq i}^{K}2\operatorname{diag}\left(\boldsymbol{C}_i^k\left(\boldsymbol{C}_i^k\right)^T\right)+2\eta\boldsymbol{I}.$$

$\nabla^2 f_i(\boldsymbol{\chi}_i)$ will be positive definite if the following matrix $\boldsymbol{S}$ is positive definite:

$$\boldsymbol{S} = \boldsymbol{N}_i \boldsymbol{N}_i^T - \left(\boldsymbol{P}_i \boldsymbol{P}_i^T + \sum_{k=1,k\neq i}^{K} \boldsymbol{C}_i^k \left(\boldsymbol{C}_i^k \right)^T \right) + \eta \boldsymbol{I} \quad .$$

After some derivations, we have

$$\boldsymbol{S} = \left(1+\eta\right)\boldsymbol{I} - \boldsymbol{E}_i^i \left(2/n_i - 2/n + \sum_{k=1}^{K} n_k / n^2 \right).$$

In order to make $\boldsymbol{S}$ positive define, each eigenvalue of $\boldsymbol{S}$ should be greater than 0. Because the maximal eigenvalue of $\boldsymbol{E}_i^i$ is n_i, we should ensure

$$\left(1+\eta\right) - n_i \left(2/n_i - 2/n + \sum_{k=1}^{K} n_k / n^2 \right) > 0$$

For $n=n_1+n_2+\ldots+n_K$, we have $\eta>1-n_i/n$, which could guarantee that $f_i(\boldsymbol{X}_i)$ is convex to $\boldsymbol{X}_i$.

Chapter 7. Conclusion

7.1 Conclusion

We have addressed in this thesis several important issues of sparse representation and dictionary learning for pattern classification, especially face recognition, by using tools from statistical machine learning, convex optimization, pattern classification, and computer vision.

First, we proposed the model of collaborative representation based classification (CRC), which has various instantiations by applying different norms to the coding residual and coding coefficient. By this model, we illustrated how SRC works and showed that the collaborative representation mechanism used in SRC is much more crucial than the l_1-norm sparsity of coding coefficients to the success of face classification. More specifically, the l_1 or l_2 norm characterization of coding residual is related to the robustness of CRC to outlier facial pixels, while the l_1 or l_2 norm characterization of coding coefficient is related to the degree of discrimination of facial features.

Second, we discussed the use of local features in sparse representation model. A Gabor feature based robust representation and classification (GRRC) scheme was proposed for robust face recognition. The use of Gabor features not only increases the face discrimination power due to the multi-scale and multi-orientation description, but also allows us to compute a compact Gabor occlusion dictionary, which has significantly smaller size than the identity occlusion dictionary. More importantly, we showed that with Gabor feature transformation, l_2-norm could take place the role of l_1-norm to regularize the coding coefficients in face classification tasks, which further reduces significantly the computational cost in coding occluded face images.

The sparse coding model with the data fidelity term measured by l_2-norm or l_1-norm actually assumes that the coding residual follows Gaussian or Laplacian distribution,

which may not be effective enough to describe the coding residual in practical FR systems. To solve this problem, we proposed a new face coding model, namely regularized robust coding (RRC), which could robustly regress a given signal with regularized regression coefficients. By assuming that the coding residual and the coding coefficient are respectively independent and identically distributed, the RRC seeks for a maximum a posterior solution of the coding problem. An iteratively reweighted regularized robust coding algorithm was proposed to solve the RRC model efficiently. Extensive experiments on representative face databases demonstrated that the RRC is much more effective and efficient than state-of-the-art sparse representation based methods in dealing with face occlusion, corruption, lighting and expression changes, etc.

Finally, we presented a novel Fisher discrimination dictionary learning (FDDL) method to fully exploit the discrimination information in coding coefficient and coding residual. Based on the Fisher discrimination criterion, a structured dictionary, whose dictionary atoms have correspondence to the class labels, was learnt so that the reconstruction error after sparse coding can be used for pattern classification. Meanwhile, the Fisher discrimination criterion was imposed on the coding coefficients so that they have small within-class scatter but big between-class scatter. A new classification scheme associated with the proposed FDDL method was then presented by using both the discriminative information in the reconstruction error and the sparse coding coefficients. The proposed FDDL was extensively evaluated on benchmark image databases in comparison with existing sparsity and DL based classification methods.

7.2 Future Work

There are several extensions of this thesis, which we are investigating and are not presented in this thesis. The first one is the extension of Chapter 3. We have proposed a relaxed collaborative representation model to exploit the similarity and distinctiveness of

features for pattern classification. The preliminary results have been published in [259]. Another interesting question is how to effectively handle the misalignment in face recognition under the sparse or collaborative representation based classification framework. We have proposed a misalignment robust representation model in [260], which has similar recognition accuracy to the state-of-the-art robust alignment by sparse representation (RASR) [189] but with much faster speed (e.g., 100 times speedup). We will further enhance the above two extensions in the future work.

Apart from the above extensions, there are still some theoretical problems on collaborative representation and sparse regularization to be more deeply investigated, such as how to effectively exploit the inspiration of sparse coding mechanism of human vision system for pattern classification, how sparsity brings distinctiveness for pattern classification, why l_2-norm regularized collaborative representation could achieve similar performance as sparse representation, and why the SRC/CRC classifier works better than the nearest neighbor and nearest subspace classifiers. Although some empirical discussions on them have been made in Chapter 3, more theoretical supports from statistical machine learning theory and biological vision need to be further studied.

Dictionary learning is also a very important topic in this thesis. Recently, dictionary learning for labeled and unlabeled data, dictionary learning with classifier training, and dictionary learning with complex regularization (e.g., structured and hierarchical sparsity) have been attracting much attentions from researchers. We will make further studies along these directions as the following work of Chapter 6.

In this thesis, the training samples are all labeled and all the proposed methods employ an automatic learning scheme. However, in some practical applications, it may be difficult, time-consuming, or expensive to obtain labeled samples. In the future, we will consider the classification problem where only a part of training samples are labeled. In such case, the semi-supervised learning or active learning techniques could be adopted.

Finally, more general and practical problems, such as image classification, object recognition, texture recognition, and image reconstruction, could be applied by adapting

the outputs of this thesis. We believe that these new applications will open up many possibilities in the exploration of designing novel classification and learning methods based on sparse or collaborative representation.

Bibliography

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Survey*, vol. 35, no. 4, pp. 399-458, 2003.

[2] S. Z. Li and A. K. Jain, *Handbook of Face Recognition (Second Edition).* Springer, 2011.

[3] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," *Journal of Information Processing Systems*, vol.5, no.2, pp. 41-68, 2009.

[4] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4-20, 2004.

[5] H. Shiono, "Personal identification using DNA polymorphism—the identification of forensic biological materials," *Nihon Hoigaku Zasshi*, vol. 50, no. 5, pp. 320-330, 1996.

[6] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148-1161, 1993.

[7] J. G. Daugman, "How iris recognition works," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21-30, 2004.

[8] K. Miyazawa, K. Ito, T. Aoki, K. Kobayashi, and H. Nakajima, "An effective approach for iris recognition using phase-based image matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1741-1756, 2008.

[9] Z. N. Sun and T. N. Tan, "Ordinal measures for iris recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2211-2226, 2009.

[10] K. W. Bowyer, K. Hollingsworth, and P. J. Flynn, "Image understanding for iris biometrics: a survey," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 281-307, 2008.

[11] R. B. Hill, *"Retinal Identification," in Biometrics: Personal Identification in Networked Society*. A. K. Jain, R. Bolle, and S. Pankanti, Eds., Kluwer Academic, 1999.

[12] H. Borgen, P. Bours, and S. D. Wolthusen, "Visible-spectrum biometric retina recognition," In *Proc. Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, 2008.

[13] P. Yan and K. W. Bowyer, "Biometric recognition using 3D ear shape," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1297-1308, 2007.

[14] B. Bhanu and H. Chen, *Human Ear Recognition by Computer.* Springer, 2008.

[15] L. Hong, Y. F. Wan, and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777-789, 1998.

[16] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition (second edition)*. Springer, 2009.

[17] N. Ratha and R. Bolle, *Automatic Fingerprint Recognition Systems*. Springer, 2004.

[18] J. W. Gu, J. Zhou, and C. Y. Yang, "Fingerprint recognition by combining global structure and local cues," *IEEE Trans. Image Processing*, vol. 15, no. 7, pp. 1952-1964, 2006.

[19] Q. J. Zhao, D. Zhang, L. Zhang, and N. Luo, "High resolution partial fingerprint alignment using pore-valley descriptors," *Pattern Recognition*, vol. 43, no. 3, pp. 1050-1061, 2010.

[20] L. Zhang, L. Zhang, H. L. Zhu, and D. Zhang, "Online finger-knuckle-print verification for personal authentication," *Pattern Recognition*, vol. 43, no. 7, pp. 2560-2571, 2010.

[21] L. Zhang, L. Zhang, H. L. Zhu, and D. Zhang, "Ensemble of local and global information for finger-knuckle-print recognition," *Pattern Recognition*, vol. 44, no. 9, pp. 1990-1998, 2011.

[22] A. K. Jain, A. Ross, and S. Pankanti, "A prototype hand geometry-based verification system," In *Proc. Int'l Conf. Audio and Video-based Biometric Person Authentication*, 1999.

[23] N. Duta, "A survey of biometric technology based on hand shape," *Pattern Recognition*, vol. 42, no. 11, pp. 2797-2806, 2009.

[24] J. P. Campbell, "Speaker recognition: a tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437-1462, 1997.

[25] H. Hollien, *Forensic Voice Identification.* Academic Press, 2001.

[26] R. Plamondon and G. Lorette, "Automatic signature verification and writer identification — the state of the art," *Pattern Recognition*, vol. 22, no. 2, pp. 107-131, 1989.

[27] C. N. Liu, N. M. Herbst, and N. J. Anthony, "Automatic signature verification: system description and field test results," *IEEE Trans. Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 35-38, 1979.

[28] D. Zhang, W-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041-1050, 2003.

[29] L. Zhang and D. Zhang, "Characterization of palmprints by wavelet signatures via directional context modeling," *IEEE Trans. System, Man and Cybernetics B*, vol. 34, no. 3, pp. 1335-1347, 2004.

[30] Z. N. Sun, T. N. Tan, Y. H. Wang, and S. Z. Li, "Ordinal palmprint representation for personal identification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005.

[31] A. Kong, D. Zhang, and M. Kamel, "A survey of palmprint recognition," *Pattern Recognition*, vol. 42, no. 7, pp. 1408-1418, 2009.

[32] A. K. Jain and J. J. Feng, "Latent palmprint matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1032-1047, 2009.

[33] D. Zhang, G. M. Lu, W. Li, L. Zhang, and N. Luo, "Palmprint recognition using 3-D information," *IEEE Trans. System, Man and Cybernetics C*, vol. 39, no. 5, pp. 505-519, 2009.

[34] W. Li, L. Zhang, D. Zhang, G. M. Lu, and J. Q. Yan, "Efficient joint 2D and 3D palmprint matching with alignment refinement," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[35] D. Zhang, Z. H. Guo, G. M. Lu, L. Zhang, and W. M. Zuo, "An online system of multispectral palmprint verification," *IEEE Trans. Instrumentation and Measurement*, vol. 59, no. 2, pp. 480-490, 2010.

[36] M. S. Nixon, T. N. Tan, and R. Chellappa, *Human Identification Based on Gait.* Springer, 2006.

[37] FaceIt-Hist, http://www.identix.com/company/comp_history.html.

[38] C. G. Tredoux, Y. Rosenthal, L. d. Costa, and D. Nunez, "Face reconstruction using a configural, eigenface-based composite system," In *Proc. 3rd Biennial Meeting of the Society for Applied Research in Memory and Cognition (SARMAC)*, 1999.

[39] Facebook facial-recognition tagger goes live, http://blogs.wsj.com/digits/2009/11/11/facebook-facial-recogniton-tagger-goes-live.

[40] P. Y. Hong, Z. Wen, T. S. Huang, "Iface: a 3D synthetic talking face," *Int. J. Image Graph*, vol. 1, no. 1, pp. 19-26, 2001.

[41] J. S. Bruner and R. Tagiuri, "The perception of people," In *G. Lindzey (Ed.), Handbook of Social Psychology*, vol. 2, pp. 634-654, 1954. Reading, Massachusetts: Addison-Wesley.

[42] W. W. Bledsoe, "*The model method in facial recognition*," Tech. rep. PRI: 15, Panoramic research Inc., Palo Alto, CA, 1964.

[43] T. Kanade, "*Picture processing by computer complex and recognition of human faces*," Tech. Rep., Kyoto Univ., Dept. Inform. Sci., 1973.

[44] R. Brunelli, T. Poggio, "Face recognition: Features versus templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 10, pp. 1042-1052, 1993.

[45] P. J. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The FERET evaluation methodology for face recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090-1104, 2000.

[46] D. Blackburn, M. Bone, and P. J. Phillips, "*Face recognition vendor test 2000*," Tech. Rep., 2001, http://www.frvt.org.

[47] P. J. Phillips, P. J. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, and J. M. Bone, "*Face recognition vendor test 2002*," Evaluation report, NISTIR 6965, 2003, http://www.frvt.org.

[48] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe, "FRVT 2006 and ICE 2006 large-scale experimental results," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 831-846, 2010.

[49] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitres, "XM2VTSDB: The extended M2VTS database," In *Proc. Int'l conf. Audio- and Video- Based Person Authentication*, 1999.

[50] X. Y. Tan, S. C. Chen, Z. H. Zhou, and F. Y Zhang, "Face recognition from a single image per person: A survey," *Pattern Recognition*, vol. 39, pp. 1725-1745, 2006.

[51] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: A survey," *Pattern Recognition Letter*, vol. 28, no. 14, pp. 1885-1906, 2007.

[52] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 705-741, 1995.

[53] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi, "Recent advances in visual and infrared face recognition-a review," *Computer Vision and Image Understanding*, vol. 97, no. 1, pp. 103-135, 2005.

[54] B. Klare and A. K. Jain, "Heterogeneous face recognition: Matching NIR to visible light images," In *Proc. Int'l Conf. Pattern Recognition*, 2010.

[55] X. Wang and X. Tang, "Face photo - sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 31, no. 11, pp. 1955-1967, 2009.

[56] X. Wang, and X. Tang, "A unified framework for subspace face recognition," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 26, no. 9, pp. 1222-1228, 2004.

[57] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.

[58] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriengman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 19, no. 7, pp. 711-720, 1997.

[59] P. Comon, "Independent component analysis —a new concept?" *Signal Process*, vol. 36, no. 3, pp. 287-314, 1994.

[60] C. Jutten, and J. Herault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture," *Signal Process*, vol. 24, no. 1, pp. 1-10, 1991.

[61] B. Moghaddam, and A. Pentland, "Probabilistic visual learning for object detection," In *Proc. Int'l Conf. Computer Vision*, 1995.

[62] B. Moghaddam, and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 696–710, 1997.

[63] B. Moghaddam, "Principle manifolds and probabilistic subspace for visual recognition," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 24, no. 6, pp. 780-788, 2002.

[64] B. Moghaddam, T. Jebara, and A. Pentland, "Bayesian face recognition," *Pattern Recognition*, vol. 33, no. 11, pp. 1771-1782, 2000.

[65] M.A.O. Vasilescu, and D. Terzopoulos, "Multilinear subspace analysis of image ensembles," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003.

[66] M.A.O. Vasilescu, and D. Terzopoulos, "Multilinear analysis of image ensembles: TensorFaces," In *Proc. European Conf. Computer Vision*, 2002.

[67] J. B. Tenenbaum, V. deSilva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319-2323, 2000.

[68] S. T. Roweis and L. K. Saul, "nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323-2325, 2000.

[69] T. Hastie, W. Stuetzle, "Principal curves," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 502–516, 1989.

[70] B. Schölkopf, A. Smola, K.-R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1998.

[71] M.-H. Yang, "Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods," In *Proc IEEE Int'l Conf. Automatic Face and Gesture Recognition*, 2002.

[72] T. F. Cootes, C. J. Taylor, D. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38-59, 1995.

[73] A. Hill, T. F. Cootes, and C. J. Taylor, "Active shape models and the shape approximation problem," In *Proc. British Machine Vision Conference*, 1995.

[74] X. Gao, Y. Su, X. Li, and D. Tao, "A review of active appearance models," *IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev.*, vol.40, no.2, pp.145–158, 2010.

[75] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," In *Proc. British Machine Vision Conference*, 1998.

[76] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, 2001.

[77] A. U. Batur and M. H. Hayes, "Adaptive active appearance models," *IEEE Trans. Med. Imaging*, vol. 14, no. 11, pp. 1707–1721, 2005.

[78] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775-779, 1997.

[79] B. Amberg, R. Knothe, and T. Better, "Expression invariant 3D face recognition with a morphable model," In *Proc. Int'l Conf. Automatic Face and Gesture Recognition*, 2008.

[80] V. Blanz, S. Romdhani, and T. Vetter, "Face identification across different poses and illuminations with a 3D morphable model," In *Proc. Int'l Conf. Automatic Face and Gesture Recognition*, 2002.

[81] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," In *Proc. IEEE Conf. Advanced Video and Signal Based Surveillance*, 2009.

[82] N. Ramanathan and R. Chellappa, "Modeling age progression in young faces," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006.

[83] A. Lanitis, C. J. Taylor, T. F. Cootes, "Toward automatic simulation of aging effects on face images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 442–455, 2002.

[84] U. Park, Y. Tong, A. K. Jain, "Age invariant face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 947–954, 2010.

[85] T. P. Zhang, Y. Y. Tang, B. Fang, Z. W. Shang, and X. Y. Liu, "Face recognition under varying illumination using gradientfaces," *IEEE Trans. Image Processing*, vol. 18, no. 11, pp. 2599-2606, 2009.

[86] G. Tzimiropoulos, S. Zafeiriou and M. Pantic, "Principal component analysis of image gradient orientations for face recognition," In *Proc. Int'l Conf. Automatic Face and Gesture Recognition Workshops*, 2011.

[87] G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "Sparse representations of image gradient orientations for visual recognition and tracking," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, 2011.

[88] G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "Subspace learning from image gradient orientation," To appear in *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012.

[89] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 467-476, 2002.

[90] A. Timo, H. Abdenour, and P. Matti, "Face recognition with local binary patterns," In *Proc. European Conf. Computer Vision*, 2004.

[91] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, "Local gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition," In *Proc. IEEE Conf. Computer Vision*, 2005.

[92] W. Zhang, S. Shan, X. Chen, and W. Gao, "Are gabor phases really useless for face recognition?" In *Proc. Int'l Conf. Pattern Recognition*, 2006.

[93] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of gabor phase patterns (HGPP): A novel object representation approach for face recognition," *IEEE Trans. Image Processing*, vol. 16, no. 1, pp. 57-68, 2006.

[94] S. F. Xie, S. G. Shan, X. L. Chen, and J. Chen, "Fusing local patterns of gabor magnitude and phase for face recognition," *IEEE Trans. Image Processing*, vol. 19, no. 5, pp. 1349-1361, 2010.

[95] M. Yang, L. Zhang, L. Zhang and D. Zhang, "Monogenic binary pattern (MBP): A novel feature extraction and representation model for face recognition," In *Proc. Int'l Conf. Pattern Recognition*, 2010.

[96] S.Z. Li and J. Lu, "Face recognition using nearest feature line method," *IEEE Trans. Neural Network*, vol. 10, no. 2, pp. 439-443, 1999.

[97] J. T. Chien, and C. C. Wu, "Discriminant waveletfaces and nearest feature classifiers for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 12, pp. 1644-1649, 2002.

[98] J. Laaksonen, "Local subspace classifier", In *Proc. Int'l Conf. Artificial Neural Networks*, 1997.

[99] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, 2005.

[100] S.Z. Li, "Face recognition based on nearest linear combinations," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1998.

[101] I. Naseem, R. Togneri, and M. Bennamoun, "Linear regression for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 2106-2112, 2010.

[102] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, 2009.

[103] E. Elhamifar and R. Vidal, "Robust classification using structured sparse representation," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[104] Y. N. Liu, F. Wu, Z. H. Zhang, Y. T. Zhuang, and S. C. Yan, "Sparse representation using nonnegative curds and whey," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[105] X. T. Yuan and S. C. Yan, "Visual classification with multitask joint sparse representation," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[106] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "*Labeled faces in the wild: A database for studying face recognition in unconstrained environments*," Technical Report Technical Report 07-49, University of Massachusetts, 2007.

[107] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classiers for face verication," In *Proc. Int'l Conf. Computer Vision*, 2009.

[108] E. Nowak, and F. Jurie, "Learning visual similarity measures for comparing never seen objects," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2007.

[109] G. B. Huang, V. Jain, and E. Learned-Miller, "Unsupervised joint alignment of complex images," In *Proc. Int'l Conf. Computer Vision*, 2007.

[110] Q. Yin, X. O. Tang, and J. Sun, "An associate-predict model for face recognition," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[111] Overview of Face Recognition Grand Challenge (FRGC), http://www.nist.gov/itl/iad/ig/frgc.cfm

[112] D. H. Lin and X. O. Tan, "Recognize high resolution faces: From macrocosm to microcosm," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006.

[113] D. Field, "What is the goal of sensory coding?" *Neural Computation*, vol. 6, no. 4, pp. 559-601, 1994.

[114] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[115] W. E. Vinje and J. L. Gallant, "Sparse coding and decorrelation in primary visual cortex during natural vision," *Science*, vol. 287, no. 5456, pp. 1273-1276, 2000.

[116] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties," *Letter to Nature*, vol. 381, pp. 607-609, 1996.

[117] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267-288, 1996.

[118] P. Zhao and B. Yu, "On model selection consistency of LASSO," *J. Machine Learning Research*, no. 7, pp. 2541-2563, 2006.

[119] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129-159, 2001.

[120] D. Donoho and M. Elad, "Optimal sparse representation in general (non-orthogoinal) dictionaries via l_1 minimization", In *Proc. National Academy of Sciences*, vol. 100, no. 5, pp. 2197-2202, 2003.

[121] 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.*, vol. 59, no. 6, pp. 797-829, 2006.

[122] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. on Information Theory*, vol. 51, no. 12, pp. 4203-4215, 2005.

[123] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[124] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," In *Proc. IEEE Special Issue on Applications of Compressive Sensing & Sparse Representation*, vol. 98, no. 6, pp. 948-958, 2010.

[125] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," In *Proc. 27th Annual Asilomar Conf. Signals, Syst. Comput.*, 1993.

[126] A.Y. Yang, A. Ganesh, Z.H. Zhou, S.S. Sastry, and Y. Ma, "*A review of fast l_1-minimization algorithms for robust face recognition*," arXiv:1007.3753v2, 2010.

[127] E. Candès, "Compressive sampling," In *Proc. Int. Congress of Mathematics*, 2006.

[128] J. Bobin, J. Starck, J. Fadili, Y. Moudden, and D. Donoho, "Morphological component analysis: An adaptive thresholding strategy", *IEEE Trans. on Image processing*, vol. 16, no. 11, pp. 2675-2681, 2007.

[129] M. Aharon, M. Elad, and A.M. Bruckstein, "The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Processing*, vol. 54, no. 11, pp. 4311-4322, 2006.

[130] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, "Non-local sparse models for image restoration," In *Proc. Int'l Conf. Computer Vision*, 2009.

[131] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[132] I. Daubechies, M. Defriese, and C. DeMol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413-1457, 2004.

[133] J. M. Bioucas Dias, and M.A.T. Figueiredo. "A new TwIST: two-step iterative shrinkage /thresholding algorithms for image restoration," *IEEE Trans. on Image Processing*, vol.16, no.12, pp. 2992-3004, 2007.

[134] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Trans. on Image Processing*, vol. 18, no. 1, pp. 27-36, 2009.

[135] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[136] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *SIAM Multiscale Modeling and Simulation*, vol. 7, no. 1, pp. 214-241, 2008.

[137] J. C. Yang, J. Wright, Y. Ma, and T. Huang, "Image super-resolution as sparse representation of raw image patches," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[138] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. on Image Processing*, vol. 20, no. 7, pp. 1838-1857, 2011.

[139] M. Elad, M.A.T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," In *Proc IEEE Special Issue on Applications of Compressive Sensing & Sparse Representation*, vol. 98, no. 6, pp. 972-982, 2010.

[140] E. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. On Information Theory*, vol. 52, no. 12, pp. 5406 - 5425, 2006.

[141] P. L. Combettes, and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM J. Multiscale Model. Simul.*, vol. 4, no. 4, pp.1168-1200, 2005.

[142] E. Cande`s, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure and Applied Math.*, vol. 59, no. 8, pp. 1207-1223, 2006.

[143] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, vol. 52, no. 2, pp. 489 - 509, 2006.

[144] K. Huang and S. Aviyente, "Sparse representation for signal classification," In *Proc. Neural Information Processing Systems*, 2006.

[145] F. Rodriguez and G. Sapiro, "*Sparse representation for image classification: Learning discriminative and reconstructive non-parametric dictionaries*," IMA Preprint 2213, 2007.

[146] Face recognition homepage, http://www.face-rec.org/.

[147] S. H. Gao, I. W-H. Tsang, and L-T. Chia, "Kernel sparse representation for image classification and face recognition," In *Proc. European Conf. Computer Vision*, 2010.

[148] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045-1057, 2010.

[149] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp.2037–2041, 2006.

[150] J. C. Yang, J. Wright, T. Huang and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861-2873, 2010.

[151] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," *Journal of Visual Communication and Image Representation*, vol. 19, no. 4, pp. 270-282, 2008.

[152] P. Sprechmann and G. Sapiro, "Dictionary learning and sparse coding for unsupervised clustering," In *Proc. Int'l Conf. Acoustics Speech and Signal Processing*, 2010.

[153] J. Mairal, M. Leordeanu, F. Bach, M. Hebert and J. Ponce, "Discriminative sparse image models for class-specific edge detection and image interpretation," In *Proc. European Conf. Computer Vision*, 2008.

[154] B. J. Shastri, and M. D. Levin, "Face recognition using localized features based on non-negative sparse coding," *Mach. Vision Appl.*, vol. 18, no. 2, pp. 107-122, 2007.

[155] D. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[156] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, "Supervised dictionary learning," In *Proc. Neural Information Processing Systems*, 2009.

[157] Q. Zhang and B.X. Li, "Discriminative K-SVD for dictionary learning in face recognition," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[158] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[159] J. C. Yang, K. Yu, and T. Huang, "Supervised translation-invariant sparse coding," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[160] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," In *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, 1999.

[161] K. Engan, B. D. Rao, and K. Kreutz-Delgado, "Frame design using FOCUSS with method of optimal directions (MOD)," In *Proc. Norwegian Signal Process. Symp.*, 1999.

[162] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zissserman, "Learning discriminative dictionaries for local image analysis," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[163] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, pp. 19-60, 2010.

[164] Patrik O. Hoyer, "Non-negative sparse coding," In *Proc. IEEE Workshop Neural Networks for Signal Processing*, 2002.

[165] S. Bengio, F. Pereira, Y. Singer, and D. Strelow, "Group sparse coding," In *Proc. Neural Information Processing Systems*, 2009.

[166] Francis R. Bach, "Consistency of the group lasso and multiple kernel learning," *Journal of Machine Learning Research*, vol. 9, pp. 1179-1225, 2008.

[167] L. Jacob, G. Obozinski, and J. Vert, "Group lasso with overlap and graph lasso," In *Proc. Int'l Conf. Machine Learning*, 2009.

[168] H. Liu, M. Palatucci, and J. Zhang, "Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery," In *Proc. Int'l Conf. Machine Learning*, 2009.

[169] G. Obozinski, B. Taskar, and M. Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems," *Journal of Statistics and Computing*, vol. 20, no. 2, pp. 231–252, 2010.

[170] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach, "Proximal methods for hierarchical sparse coding," *Journal of Machine Learning Research*, vol. 12, pp. 2297-2334, 2011.

[171] S. Kim, and E. P. Xing, "Tree-guided group lasso for multi-task regression with structured sparsity," In *Proc. Int'l Conf. Machine Learning*, 2010.

[172] J. Z. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," In *Proc. Int'l Conf. Machine Learning*, 2009.

[173] P. Vincent and Y. Bengio, "K-local hyperplance and convex distance nearest neighbor algorithms," In *Proc. Neural Information Processing Systems*, vol. 14, pp. 985-992, 2001.

[174] Y. Adini, Y. Moses, and S. Ullman, "Face recognition: The problem of compensating for changes in illumination direction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 721–732, 1997.

[175] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," In *Proc. Int'l Conf. Computer Vision*, 2001.

[176] P. Belhumeur and D. Kriegman, "What is the set of images of an object under all possible lighting conditions," *Int. J. Computer Vision*, vol. 28, no. 3, pp. 245–260, 1998.

[177] R. Ramamoorthi and P. Hanrahan, "An efficient representation for irradiance environment," In *Proc. Annual conf. Computer graphics and interactive techniques*, 2001.

[178] R. Ramamoorthi and P. Hanrahan, "A signal-processing framework for inverse rendering," In *Proc. Annual conf. Computer graphics and interactive techniques*, 2001.

[179] L. Zhang, M. Yang, X. C. Feng, and D. Zhang, "Collaborative representation based classification for face recognition," *IEEE Trans. on Image Processing* (submitted).

[180] L. Zhang, M. Yang, X. C. Feng and D. Zhang, "Sparse representation or collaborative representation which helps face recognition?" In *Proc. Int'l Conf. Computer Vision*, 2011.

[181] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with gabor occlusion dictionary," In *Proc. European Conf. Computer Vision*, 2010.

[182] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. Huang, "Learning with l_1-graph for image analysis," *IEEE Trans. Image Processing*, vol. 19, no. 4, pp. 858-866, 2010.

[183] L. S. Qiao, S. C. Chen, and X. Y. Tan, "Sparsity preserving projections with applications to face recognition", *Pattern Recognition*, vol. 43, no. 1, pp. 331-341, 2010.

[184] Z. L. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[185] M. Yang, L. Zhang, X. C. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," In *Proc. Int'l Conf. Computer Vision*, 2011.

[186] J.-L. Starck, M. Elad, and D. L. Donoho. "Redundant multiscale transforms and their application for morphological component analysis," *Journal of Advances in Imaging and Electron Physics*, vol. 132, pp. 287-348, 2004.

[187] Z. Zhou, A. Wagner, H. Mobahi. J. Wright, and Y. Ma, "Face recognition with contiguous occlusion using markov random fields," In *Proc. Int'l Conf. Computer Vision*, 2009.

[188] J. Z. Huang, X. L. Huang, and D. Metaxas, "Simultaneous image transformation and sparse representation recovery," In *Proc. IEEE Conf. Computer Vision and Pattern Recognitio*n, 2008.

[189] 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 *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[190] J. Yang, K. Yu, Y. Gong and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[191] Y. G. Peng, A. Ganesh, J. Wright, W. L. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[192] S. H. Gao, I. W-H. Tsang, L-T. Chia, and P. L. Zhao, "Local features are not lonely-laplacian sparse coding for image classification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[193] P. Berkes, B. L. White, and J. Fiser, "No evidence for active sparsification in the visual cortex," In *Proc. Neural Information Processing Systems*, 2009.

[194] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Regularized robust coding for face recognition," *IEEE Transactions on Image Processing* (Submitted).

[195] M. Yang, L. Zhang, J. Yang and D. Zhang, "Robust sparse coding for face recognition," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[196] J. Mairal, F. Bach and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*. vol. 34, no. 4, pp. 791-804, 2012.

[197] M. Yang, L. Zhang, X. C. Feng, and D. Zhang, "*Fisher discrimination dictionary learning for sparse representation,*" Prepared.

[198] Q. F. Shi, H. X. Li and C. H. Shen, "Rapid face recognition using hashing," In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2010.

[199] J. B. Huang and M. H. Yang, "Fast sparse representation with prototypes," In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2010.

[200] S. J. Wright, R. D. Nowak, and M.A.T. Figueiredo, "Sparse reconstruction by separable approximation," In *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing*, 2008.

[201] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM. J. Imaging Science*, vol. 2, no. 1, pp. 183-202, 2009.

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

[203] D. Malioutove, M. Cetin, and A. Willsky, "Homotopy continuation for sparse signal representation," In *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing*, 2005.

[204] 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*, vol. 1, no. 4, pp. 606–617, 2007.

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

[206] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, 2001.

[207] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005.

[208] S. Kindermanny, S. Osherz, and P.W. Jones, "*Deblurring and Denoising of Images by Nonlocal Functionals,*" http://www.math.ucla.edu/applied/cam/index.html, vol. 04–75, UCLA-CAM Rep., 2004.

[209] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Trans. Image Processing*, vol. 18, no. 1, pp. 36-51, 2009.

[210] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 2003.

[211] Z. Lin, M. Chen, L. Wu, and Y. Ma, "*The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices,*" UIUC Technical Report UILU-ENG-09-2215, 2009.

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

[213] R. Gross, I. Matthews. J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[214] John Wright, Arvind Ganesh, Allen Yang, Zihan Zhou, Yi Ma, "*Sparsity and robustness in face recognition*," arXiv:1111.1014v1, 2011.

[215] A.Y. Yang, S. S. Sastry, A. Ganesh, and Yi Ma, "Fast l_1-minimization algorithms and an application in robust face recognition: A review," In *Proc. IEEE Int'l Conf. Image Processing*, 2010.

[216] M. Kirby and L. Sirovich, "Application of the KL procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 1, pp.103-108, 1990.

[217] J. Yang, J.Y. Yang, "Why can LDA be performed in PCA transformed space?" *Pattern Recognition*, vol. 36, no. 2, pp. 563-566, 2003.

[218] X. He, S. Yan, Y. Hu, P. Niyogi, and H. J. Zhang, "Face recognition using laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328-340, 2005.

[219] H.-T.Chen, H.-W. Chang, and T.-L. Liu, "Local discriminant embedding and its variants," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 846-853, 2005.

[220] J. Yang, D. Zhang, J.-Y. Yang, and B. Niu, "Globally maximizing, locally minimizing: Unsupervised discriminant projection with applications to face and palm biometrics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 650-664, 2007.

[221] D. Gabor, "Theory of communication," *J. Inst. Elect. Eng.*, vol. 93, no. 26, pt. III, pp. 429-457, 1946.

[222] J.P. Jones and L.A. Palmer, "An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex", *Journal of Neurophysiology*, vol. 58, no. 6, pp. 1233-1258, 1987.

[223] M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C.v.d. Malsburg, R.P. Würtz and W. Konen, "Distortion invariant object recognition in the dynamic link architecture", *IEEE Transactions on Computers*, vol. 42, no. 3, pp. 300-311, 1993.

[224] L. Shen and L. Bai, "A review on Gabor wavelets for face recognition," *Pattern Analysis and Application*, vol. 9, no. 10, pp. 273-292, 2006.

[225] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 2, pp. 218-233, 2003.

[226] E. Candès and J. Romberg, "*L1-magic: A collection of MATLAB routines for solving the convex optimization programs central to compressive sampling*," 2006 [Online], Available: www.acm.cal tech.edu/l1magic/.

[227] The MOSEK Optimization Tools Version 2.5. User's Manual and Reference 2002 [Online]. Available: www.mosek.com,MOSEK.

[228] Y. Nesterov, A. Nemirovskii, "Interior-point polynomial algorithms in convex programming," *SIAM* Philadelphia, PA, 1994.

[229] P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *Image and Vision Computing*, vol. 16, No. 5, pp. 295-306, 1998.

[230] A. Leonardis and H. Bischof, "Robust recognition using eigenimages," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 99-118, 2000.

[231] S. Chen, T. Shan, and B.C. Lovell, "Robust face recognition in rotated eigenspaces," In *Proc. Int'l Conf. Image and Vision Computing*, 2007.

[232] A.M. Martinez, "Recognizing Imprecisely localized, partially occluded, and expression variant faces from a single sample per class," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 748-763, 2002.

[233] H. Jia and A. Martinez, "Support vector machines in face recognition with occlusions," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[234] J. Wright and Y. Ma, "Dense error correction via l_1 minimization," *IEEE Trans. Information Theory*, vol. 56, no. 7, pp. 3540–3560, 2010.

[235] I. Daubechies, R. Devore, M. Fornasier, and C.S. Gunturk, "*Iteratively re-weighted least squares minimization for sparse recovery*," arXiv: 0807.0575.

[236] E.J. Candès, M.B. Wakin, and S.P. Boyd, "Enhancing sparsity by reweighted l_1 minimization," *J. Fourier Anal. Appl.*, vol.14, pp. 877-905, 2008.

[237] J.J. Wang, J.C. Yang, K. Y, F.J Lvx, T. Huangz, and Y.H. Gong, "Locality-constrained linear coding for image classification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.

[238] R. Rigamonti, M. Brown, V. Lepetit, "Are sparse representations really relevant for image classification?" In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[239] R. He, W.S. Zheng, B.G. Hu, and X.W. Kong, "A regularized correntropy framework for robust pattern recognition," *Neural Computation*, vol. 23, pp. 2074-2100, 2011.

[240] R. He, W.S. Zheng, and B.G. Hu, "Maximum correntropy criterion for robust face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1561-1576, 2011.

[241] W.F. Liu, P.P. Pokharel, and J.C. Principe, "Correntropy: properties and applications in non-gaussian signal processing," *IEEE Trans. Signal Processing*, vol. 55, No. 11, pp. 5286-8298, 2007.

[242] P. Huber. *Robust Statistics.* New York: Wiley, 1981.

[243] P.J. Huber, "Robust regression: Asymptotics, conjectures and Monte Carlo," *Ann. Stat.*, vol. 1, no. 5, pp. 799–821, 1973.

[244] Z.Y. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting," *Image and Vision Computing*, vol.15, no.1, pp. 59-76, 1997.

[245] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer Series in Statistics, 2003.

[246] M.I. Jordan, "*Why the logistic function? A tutorial discussion on probabilities and neural networks*," MIT Computational Cognitive Science Report 9503, 1995.

[247] J. Hiriart-Urruty and C. Lemarechal, *Convex analysis and minimization algorithms*. Springer-Verlag, 1996.

[248] J.R. Shewchuk, "*An introduction to the conjugate gradient method without the agonizing pain*," (Tech. Rep. CMU-CS-94-125) Pittsburgh, PA: School of Computer Science, Carnegie Mellon University.

[249] P. Viola and M.J. Jones, "Robust real-time face detection," *Int'l J. Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.

[250] S. Fidler, D. Skocaj, and A. Leonardis, "Combining reconstructive and discriminative subspace methods for robust classification and regression by subsampling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 337-350, 2006.

[251] M. Yang, L. Zhang, J. Yang and D. Zhang, "Metaface learning for sparse representation based face recognition," In *Proc. IEEE Int'l Conf. Image Processing*, 2010.

[252] R. Duda, P. Hart, and D. Stork, *Pattern Classification (2nd ed.)*. Wiley-Interscience, 2000.

[253] L. Rosasco, A. Verri, M. Santoro, S. Mosci, and S. Villa, "*Iterative projection methods for structured sparsity regularization*," MIT Technical Reports, MIT-CSAIL-TR-2009-050, CBCL-282, 2009.

[254] USPS Handwritten Digit Database. http://www-i6. informatik.rwth-aachen.de/~keysers/usps.html.

[255] M. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006.

[256] M. Nilsback and A. Zisserman, "Automated flower classification, over a large number of classes," In *Proc. IEEE Conf. Computer Vision, Graphics & Image Processing*, 2008.

[257] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," In *Proc. Int'l Conf. Computer Vision*, 2009.

[258] M. Yang, L. Zhang, S. C. K. Shiu, and D. Zhang, "Gabor Feature based Robust Representation and Classification for Face Recognition with Gabor Occlusion Dictionary," accepted by *Pattern Recognition*.

[259] M. Yang, L. Zhang, D. Zhang, and S. L. Wang, "Relaxed collaborative representation for pattern classification," In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012.

[260] M. Yang, L. Zhang and D. Zhang, "Misalignment robust representation for face recognition," In *Proc. European Conf. Computer Vision*, 2012.