Learning Sparse Representations for Computer Vision Applications

by

Homa Foroughi

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computing Science

University of Alberta

Abstract

At the core of many computer vision methods lies the question of how to represent data. Representing the data in a meaningful way, which highlights its most useful properties, can significantly affect the performance of any vision-based application. Traditional systems are heavily reliant on hand-designed representations that are mostly domain-specific and also need significant amounts of domain knowledge and human effort. Recently, there has been much research in learning representation from data and one of successful approaches is the sparse representation, which tries to represent data as a linear combination of a few elements of a basis or dictionary. A good sparse representation of an image is expected to have high fidelity to the observed image content and reveal its underlying structure and semantic information at the same time. In this thesis, we address the problem of how to learn such representation or dictionary from training images, particularly for crowd counting, image classification, and dimensionality reduction tasks.

Counting pedestrians in videos is a topic of great interest in areas such as visual surveillance, public resource management and security purposes. Crowd counting could be a challenging task due to severe occlusions, scene perspective distortions and diverse crowd distributions. In this thesis, we propose two methods for crowd counting based on compressed sensing and sparse representation theories, each of which is capable of resolving some of the aforementioned issues. Firstly, we present a counting method based on image retrieval framework, and also introduce a compact global image descriptor using compressed sensing theory, to estimate the crowd count. Next, we propose a crowd counting method based on sparse representation-based classification and random projection. We adopt a semi-supervised elastic-net to provide a rich training set, that can span variations under testing conditions. By exploiting the sequential information of readily available vast quantity of unlabeled data, we are able to annotate a large portion of data with just a handful of labeled images. Experiments on crowd counting benchmark datasets demonstrate the effectiveness and reliability of proposed methods, especially in large-scale datasets.

Image classification based on visual content is a challenging task, mainly because there is usually large amount of intra-class variability, arising from illumination and viewpoint variations, occlusion and corruption. In addition, many real-world vision applications are faced with the problem of high-dimensional data and small number of training samples. To address all these issues, we propose a joint learning framework, in which the subspace projection matrix, the dictionary and sparse coefficients are learned simultaneously. By incorporating competent constraints such as low-rank, incoherence and neighborhood preservation, we are able to learn discriminative and robust sparse representations of images, especially for challenging classification scenarios. Experimental results on several benchmark datasets verify the superior performance of our method for object classification of small datasets, which include considerable amount of different kinds of variation.

Feature selection is another solution to deal with high-dimensional data, and recently sparsity constraints have been utilized to select a subset of features. We propose a feature selection method based on the decision rule of dictionary learning, and integrate low-rank matrix recovery, reconstruction residuals, and row-sparsity constraints into the framework. As a result, the proposed method selects optimal subset of features simultaneously, and provides well-separated classes in the reduced space. Our method is capable of selecting discriminative features, even when the data are contaminated due to occlusion, illumination or pose variations and corruption. Extensive experiments on benchmark datasets verify the superior performance of the proposed method for feature selection, image/video classification and counting specific populations of tumor cells in microscopic images.

Preface

Research for this thesis was conducted under the supervision of Dr. Nilanjan Ray and Dr. Hong Zhang at the University of Alberta. Portions of this thesis were published as:

- Chapter 3: Foroughi, Homa, Nilanjan Ray, and Hong Zhang. "People counting with image retrieval using compressed sensing." 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014.

- Chapter 3: Foroughi, Homa, Nilanjan Ray, and Hong Zhang. "Robust people counting using sparse representation and random projection." Pattern Recognition 48.10 (2015): 3038-3052.

- Chapter 4: Homa Foroughi, Moein Shakeri, Nilanjan Ray and Hong Zhang. "Joint Feature Selection with Low-rank Dictionary Learning". In Proceedings of the British Machine Vision Conference (BMVC), pages 97.1-97.13. BMVA Press, September 2015.

- Chapter 5: Homa Foroughi, Nilanjan Ray and Hong Zhang. "Object Classification with Joint Projection and Low-rank Dictionary Learning". In arXiv preprint arXiv:1612.01594, December 2016.

To people who are struggling and not giving up

Acknowledgements

I would like to sincerely acknowledge my supervisor, Dr. Nilanjan Ray, whose guidance, expertise, understanding, and patience, added considerably to my graduate experience. I would also like to express my appreciation to my co-supervisor, Dr. Hong Zhang, for all the constructive advice and thoughtful comments throughout this research. The Ph.D. program has been an enjoyable journey for me because of their supports and guidance.

I would like to express my gratitude to my committee members, Dr. Pierre Boulanger and Dr. Martin Jagersand, and also my external examiner, Dr. Allan Jepson, for their invaluable time and constructive feedbacks that improved the technical quality of my work.

I also thank all the people who have helped me to improve the quality of my research by providing helpful comments and discussions including faculty members, fellow students, and anonymous reviewers of the research papers that I have published. This research study would not be successful without supporting grants from Centre for Intelligent Mining System (CIMS) and Alberta Innovates Technology Futures (AITF). I would also like to thank my friends and colleagues for making this journey more pleasant and memorable.

Finally, I wish to express my deep gratitude to my parents for their lifelong support and endless love. My sincere thank goes to my beloved husband, Moein, for his encouragement, support and love.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation and Problem Statement

The last decade brought computer vision to an advanced state, and the research results started to influence everybody's daily life, rather than being confined to industrial production lines or some specialized applications. At the core of many computer vision methods is the data representation, and good representation of data is crucial for the success of these methods. As a result, significant research efforts have been spent on designing good representations for different vision applications.

Traditional representations are often hand-designed, require significant amounts of domain knowledge and human effort, and do not generalize well to new domains. More recent methods have focused on learning representation from data, which can automatically adapt to various domains and provide more appealing solutions. Representing data as a linear combination of a few elements from a basis or dictionary, introduces the concept of sparsity, which has been the focus of much recent research in machine learning. The sparse representation of an image is expected to have high fidelity to the observed image content, and reveals its underlying structure and semantic information. Sparse representation methods gained interest along with the maturation of the compressed sensing field. One basic idea in compressed sensing is that most signals have a sparse representation as a linear combination of a reduced subset of signals from same space. Naturally, the signals tend to have a representation biased towards their own class, *i.e.*, the sparse representation is mainly formed from samples from its own class. This was the starting point for sparse representation-based classification (SRC), which proved to be an effective method for different classification tasks [140].

In this thesis, we utilize sparse representation to solve the crowd counting problem. Counting the number of pedestrians in videos has drawn a lot of attention because of intense demands in video surveillance, urban management and security purposes. Crowd counting is a challenging task due to severe occlusions, scene perspective distortions and diverse crowd distributions. Considering these challenges, we propose two robust crowd counting methods based on compressed sensing and sparse representation theories, and achieve superb performance, especially in large-scale pedestrian datasets [48], [49]. Although SRC scheme shows interesting results in different applications including people counting, the employed dictionary may not be effective enough to represent the query images due to the uncertain and noisy information in the original training images. Using the original training samples as the dictionary could not fully exploit the discriminative information hidden in the training samples. In addition, the computational complexity of SRC would be high, for using all the training samples as the dictionary. These drawbacks mostly can be effectively addressed by learning a smaller-sized and discriminative dictionary from training images.

Indeed, the dictionary which should faithfully and discriminatively represent the encoded signal, plays an important role in the success of sparse representation. It has been shown that learned dictionaries from training samples significantly outperform pre-defined off-the-shelf bases such as Wavelets [151]. The last few years have witnessed fast development on dictionary learning approaches and great success has been demonstrated in different computer vision applications [68], [148]. Supervised dictionary learning methods have been proposed to promote the discriminative power of the learned dictionary, by enforcing the representation coefficients and/or representation residual to be discriminative [149].

In this thesis, we investigate how dictionaries can be learned in a discriminative yet reconstructive manner, and introduce how smoothness priors can be incorporated into the learning framework. We discuss how sparse coding can be enriched by structuring the dictionary, and demonstrate how the dictionary can be optimized and learned for a specific task. We try to address these questions with a multidisciplinarity point of view, using tools from machine learning, convex optimization and computer vision. Our goal in this thesis is to propose algorithms for learning efficient and accurate dictionary and sparse representations (in the context of application), and use them to achieve state-of-the-art results for the important and challenging task of object classification.

For an object classification system, critical obstacles towards real-world applications are often caused by large intra-class variability, arising from different lightings, viewpoint and pose changes, occlusion and corruption, in limited training sets. Furthermore, in many areas of computer vision data are characterized by high-dimensional feature vectors, that are inefficient and computationally intensive, and their dimensionality should be reduced for an effective classification. To address these issues in a unified framework, we present a joint learning method in which the projection matrix, the dictionary and the coding coefficnets are learned simultaneously. By incorporating competent constraints such as low-rank, incoherence and neighborhood preservation, we are able to learn discriminative and robust representations of images, especially for challenging classification scenarios [50].

Another solution to deal with high-dimensional data is feature selection and recently, researchers have utilized sparsity constraints to perform feature selection [145], [144]. We propose a feature selection method based on the decision rule of dictionary learning, and integrate low-rank matrix recovery, reconstruction residuals, and row-sparsity constraints in the framework. The proposed method enables us to select discriminative features, even from noisy observations *i.e.,* occluded, corrupted, illumination and/or pose variations [51].

1.2 Contributions

The contributions of this thesis are as follows:

- We propose two crowd counting methods using sparsity. First, we present a counting method using image retrieval framework, and also introduce a compact global image descriptor based on compressed sensing theory, to estimate the crowd density for large-scale datasets [48]. It is followed by a more accurate method based on sparse representation-based classification and random projection [49]. We exploit a semi-supervised elastic-net to provide a rich set of training images of every class, that can span the variations under testing conditions. This would help us to prepare enough labeled training samples, with only a handful of user-labeled image frames. Experiments on crowd counting benchmark datasets demonstrate the effectiveness and reliability of both methods, especially in large-scale datasets.

- We propose a joint projection and low-rank dictionary learning method that simultaneously learns a robust projection matrix, a discriminative dictionary and sparse coding coeffients in the low-dimensional space, by incorporating low-rankness, structural incoherence and dual graph constraints [50]. The proposed method shows excellent accuracies to classify small-sized datasets with large intra-class variability, which may have high-dimensional feature vectors. To the best of our knowledge, this is the first proposed method that can handle all these issues simultaneously.

- We propose a joint feature selection method, under the integration of dictionary learning and low-rank matrix approximation methods, with a particular interest in preserving reconstructive relationship of data in the subset of selected features [51]. The proposed method shows superior performance both for feature selection and classification tasks. Moreover, we successfully adopt the proposed method for counting the number of specific tumor cells *e.g.*, Ki67 positive, which is an important index associated with the severity of breast cancer disease.

1.3 Thesis Outline

The remainder of the thesis is organized as follows: Chapter 2 provides an overview of sparse representation, dictionary learning and low-rank approximation, then discusses most popular methods and some important applications of them. Some related dimensionality reduction techniques and their connection to dictionary learning methods are also reviewed. In Chapter 3 we study the problem of people counting and present two novel counting methods [48], [49] using sparsity concept, and evaluate them on the crowd counting datasets. Chapter 4 describes our proposed feature selection method [51], and provides experimental results on feature selection, classification and tumor cell counting tasks. Chapter 5 presents a novel method [50] for object classification of small-sized datasets with large intra-class variability, and demonstrates various experiments on different datasets. Finally, the conclusion and directions for future work are discussed in Chapter 6.

Chapter 2

Background

This chapter provides relevant background and works related to our research.

2.1 Compressed Sensing

A conventional approach in sensing and data acquisition requires to measure (or sample) the source at its Nyquist rate before applying any compression or signal processing techniques. By exploiting a sparse structure either exposed naturally or hidden in the data, compressed sensing (CS) makes the source measurement possible at a substantially lower rate than the Nyquist rate, which is the twice of the highest frequency of the source [105]. Compressed sensing combines the measurement and compression of the data into one non-analytical, low-complexity encoding process governed by matrix-vector multiplications.

Suppose we have a N-dimensional signal $x \in R^N$, which could be sparsely represented in a certain domain by the transformation matrix $\Psi \in R^{N \times N}$. x is called K-sparse, if there are only K non-zero coefficients in the Ψ domain. The question, which may be raised is: if we already know the signal to be sampled is K-sparse, why we should take N measurements, where $N >> K$. The purpose of compressed sensing is to recover the sparse signal x, by taking M random measurements, which is much less than N. In order to take CS measurements, we first let $\Phi \in R^{M \times N}$ with $M << N$ denote the measurement matrix. The measurements matrix Φ should be uncorrelated with transform

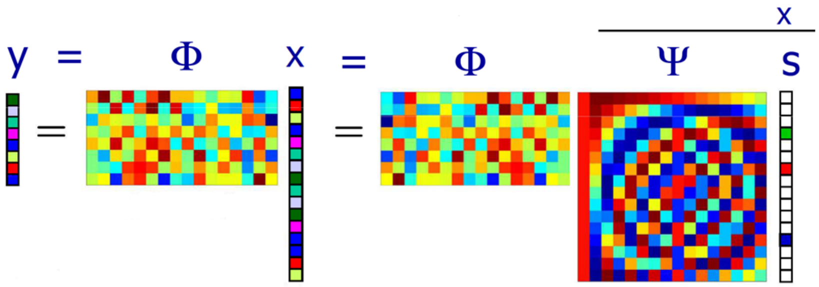


Figure 2.1: Schematic compressed sensing [8]

matrix Ψ, and satisfy the restricted isometry property (RIP) [15]:

$$(1-\delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta)\|x\|_2^2 \tag{2.1}$$

where δ is a small constant less than 1. RIP ensures that all pairwise distances between sparse signals are well preserved in the measurement space. Then the measurements are obtained by a linear system:

$$y = \Phi x \tag{2.2}$$

Suppose that the sparsification of x is given by $x = \Psi s$. Robust uncertainty principle [17] states that the signal could be exactly recovered if the number of measurements M satisfies the condition $M \geq C.K.log(N/K)$, where C is a small constant greater than 1. CS forms the under-determined linear system of equations, in which the signal is reconstructed by the following optimization:

$$\hat{s} = \underset{s}{\operatorname{argmin}} \|s\|_0 \quad s.t. \quad y = \Phi x = \Phi\Psi s \tag{2.3}$$

where $\|.\|_0$ counts the non-zero entries of a vector. The l$_0$-minimization problem is known to be NP-hard and intractable. However, developments in the CS research revealed that if the exact solution s is sufficiently sparse, the solution to the l$_0$-minimization problem can be equivalently obtained by solving the l$_1$-minimization problem as:

$$\hat{s} = \underset{s}{\operatorname{argmin}} \|s\|_1 \quad s.t. \quad y = \Phi x = \Phi\Psi s \tag{2.4}$$

where $\|s\|_1 = \sum_i |s_i|$. The l$_1$-minimization (2.4) can be solved by linear programming or greedy pursuit algorithms such as Basis Pursuit [26] or Orthogonal Matching Pursuit [108]. Once s is recovered, we can restore x from the

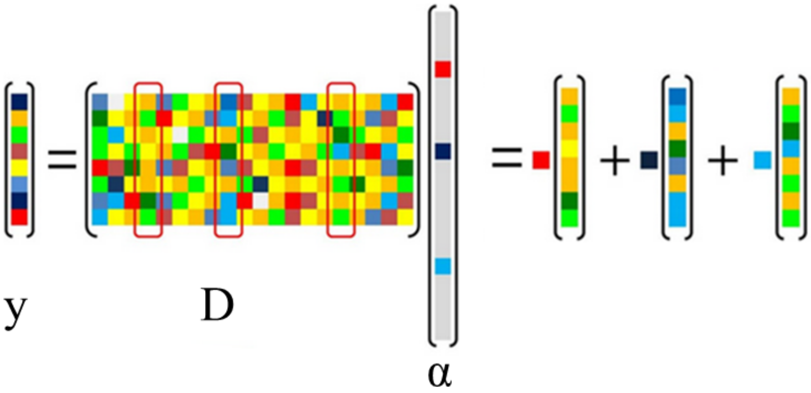


Figure 2.2: Schematic sparse coding

reverse sparsification $x = \Psi s$. RIP of Φ can guarantee a unique solution with high probability through the l$_1$-minimum decoding of s. It is known that the product $\Phi\Psi$ satisfies RIP with high probability if Φ satisfies RIP, and Ψ is orthogonal. A schematic description of CS is demonstrated in Figure 2.1.

Research in CS has focused primarily on reducing the number of measurements M, increasing robustness, and reducing the computational complexity of the recovery algorithm. The success of CS for signal reconstruction motivates the study of its potential for signal classification [140], [59]. One of the most successful applications of CS theory in computer vision and pattern recognition has been the sparse representation-based classification algorithm for face recognition [140], which is discussed in Section 2.3.

2.2 Sparse Coding

The idea of sparse coding is illustrated in Figure 2.2. According to the sparse representation theory [139], a signal $y \in R^m$ can be well-approximated by a linear combination of a few columns of some appropriate basis or an overcomplete dictionary $D \in R^{m\times n}$, that $n > m$. The representation of y may either be exact $y = D\alpha$, or approximate $y \approx D\alpha$, satisfying $\|y - D\alpha\|_2 \leq \epsilon$. The vector $\alpha \in R^n$ contains the representation coefficients of signal x. The overcomplete dictionary D makes the solution α not unique for a general case. So, the underdeterminedness is resolved by setting another constraint, known as regularization. Then, the sparsest representation is an appealing solution, which is found as:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|\alpha\|_0 \quad s.t. \quad \|y - D\alpha\|_2^2 \leq \epsilon \tag{2.5}$$

This is the familiar l_0-regularized regression problem, which is known to be intractable. One possible approach is to rely on convex relaxation that regularizes the l_1-norm of α:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|\alpha\|_1 \quad s.t. \quad \|y - D\alpha\|_2^2 \leq \epsilon \tag{2.6}$$

While the l_1-minimization problem (2.6) can be formulated as a linear program and readily solved by classical methods in convex optimization, such as interior-point methods [71], the computational complexity of those classical methods is often too high for large-scale high-dimensional image data [146]. In the light of a large number of real-world vision applications, many new efficient algorithms have been proposed over the past decade. These efficeint l_1-minimization solvers include, but not limited to, LASSO [128], Homotopy [37], Coordinate Descent algorihtm [141], Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [10], Feature-sign search algorithm [77] and Augmented Lagrangian Method (ALM) [146]. Yang *et al.* [146] provided a comprehensive comparison of some of these algorithms, and indicated that ALM and Homotopy outperform the others in terms of speed and scalability in face recognition applications.

2.3 Sparse Representation-based Classification

Sparse representation has proven to be an extremely powerful tool for acquiring, representing, and compressing high-dimensional signals and has led to promising results in many computer vision applications. This success is mainly due to the fact that a high-dimensional signal can be sparsely represented by the representative samples of its class in a low-dimensional manifold. Such

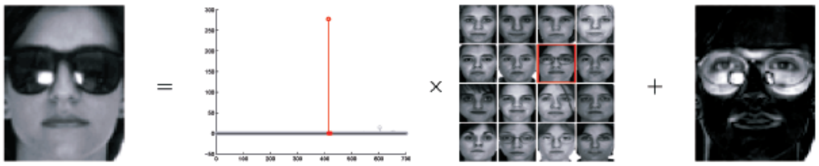


Figure 2.3: Overview of the SRC framework for face recognition [140]

a sparse representation, if computed correctly, could naturally encode the semantic information of the image [140]. Wright *et al.* [140] proposed the sparse representation-based classification (SRC) method for robust face recognition, as illustrated in Figure 2.3. Denote $X = [X_1, X_2, \ldots, X_K] \in R^{m \times n}$ the entire training set from K classes, where $X_i \in R^{m \times n_i}$ is the subset of training samples from the ith class. In the SRC, the training samples themselves are used as the dictionary. Let $y \in R^m$ a query sample, the basic steps of SRC are summarized as follows:

- **Coding**: The sparse coding coefficients of y can be obtained by solving the following l_1-minimization problem:

$$\hat{\alpha} = \underset{\alpha}{\text{argmin}} \, \|y - X\alpha\|_2^2 + \lambda\|\alpha\|_1 \tag{2.7}$$

  where λ is a scalar constant.

- **Classification**: The query sample y is assigned to the class with the smallest residual error:

$$identity(y) = \underset{i}{\text{argmin}} \, \|y - X\delta_i(\hat{\alpha})\|_2 \tag{2.8}$$

  where $\delta_i(\hat{\alpha})$ is a function that selects the coefficients associated with the i^{th} class.

The SRC classifier has a close relationship to the nearest classifiers, including the nearest neighbor (NN), nearest feature line (NFL) [84], nearest feature plane (NFP) [27], and nearest subspace (NS) [78] classifiers. The NN, NFL and NFP 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 y as the linear combination of training samples; however, one critical difference between SRC and these classifiers is that SRC collaboratively represents y by training samples from all classes, while the nearest classifiers represent y by each individual class [148].

SRC schema has shown interesting face recognition results; however, the complexity of SRC can be very high for using all the training samples, prohibiting real-time applications [68]. Equally important, the discriminative information in the training samples is not sufficiently exploited by such a naive method [150]. These problems can be addressed by learning properly a dictionary from the original training samples.

2.4 Dictionary Learning

The dictionary, which should faithfully and discriminatively represents the encoded signal, plays an important role in the success of sparse representation. There are two ways to build the dictionary: 1) building a dictionary with off-the-shelf bases that have veen designed via a analytical models such as Wavelets, Curvelets and Fourier transform, and 2) learning a dictionary from training data. Althought the analytically designed dictionaries are universal and need no training, it has been shown that learned dictionaries significantly outperform off-the-shelf bases [149] in image classification, image denoising, and so on. Most dictionary learning methods train dictionary atoms in the scheme of sparse representation regularized by l_1 sparsity constraint. The last few years have witnessed fast development on dictionary learning approaches, and great success has been demonstrated in different computer vision applications such as image restoration [97], image denoising [2], [40], face recognition [150], [68], [160], image classification [150], [149], human action recognition [150], [149], and so on.

Current prevailing dictionary learning approaches can be divided into two main categories: unsupervised and supervised. The unsupervised dictionary learning methods do not utilize class information of training samples, and their goal is to minimize the reconstruction error. The very basic model of unsupervised dictionary learning is considered as:

$$< D, A >= \underset{D,A}{\operatorname{argmin}} \|X - DA\|_F^2 \quad s.t. \quad \|a_i\|_1 \leq \epsilon \quad \forall i \tag{2.9}$$

where X is the training dataset, a_i represents a column of the coding coefficient matrix A, D is the dictionary to be learnt, and $\|.\|_F$ is the Frobenius norm. Usually, each column d_j of the dictionary is required to satisfy $\|d_j\|_2^2 \leq 1$. A common approach to minimize the above objective function alternates between two variables, minimizing w.r.t one while keeping the other fixed. The iterative solving for sparse representations based on the dictionary, and updating the dictionary given the sparse codes are performed until a stopping criterion is met. The representative unsupsevised dictionary learning methods, such as KSVD [2] and MOD [41], learn a dictionary by solving (2.9). Although these methods have achieved promising results in image restoration, they are not advantageous for image classification, since Equation (2.9) can only ensure that the learnt dictionary D could faithfully represent the training samples X. With the class labels of training samples available, the supervised dictionary learning methods exploit the class discrimination information in the learning process, which results in better classification performances [149].

In the supervised dictionary learning methods, usually additional priors on the dictionary and/or the representation coefficients are introduced in the phase of dictionary learning [148]. The general dictionary learning model in such cases is represented as:

$$< D, A >= \underset{D,A}{\operatorname{argmin}} \|X - D\,A\|_F^2 \quad s.t. \quad \begin{cases} Prior(A) \\ Prior(D) \\ \|a_i\|_1 \leq \epsilon \quad \forall i \end{cases} \tag{2.10}$$

where the constraint $Prior(A)$ introduces discrimination information for the representation coefficients, and the constraint $Prior(D)$ makes the class-specific representation residuals discriminative. In other words, the discrimination can be exploited from the coding coefficients or dictionary or both.

In the first category, a shared dictionary and a classifier over the representation coefficients are learned concurrently. In these methods, all the training samples are well reconstructed by the atoms of the shared dictionary; however, the shared dictionary loses the correspondence between the dictionary atoms and the class labels. Marial *et al.* [98] combined logistic regression with dictionary learning framework. Inspired by the KSVD algorithm, Zhang *et al.* [160] proposed a shared-dictionary learning method called discriminative KSVD (D-KSVD) for face recognition, which adds a simple linear regression to the conventional dictionary learning framework. It was followed by Jiang *et al.* [68] via adding a label consistency constraint on the coding vectors to enforce the discrimination, and the so-called LC-KSVD achieved good results in different classification tasks. Wang *et al.* [135] formulated the dictionary learning problem from a max-margin perspective and learned a dictionary using a multi-class hinge loss function. Recently, Cai *et al.* [14] formulated the discrimination term as a weighted summation of the squared distances between all pairs of coding vectors and proposed a support vector guided dictionary learning (SVGDL) model.

Let X be a set of m-dimensional n input images, *i.e.*, $X = [x_1, x_2, \ldots, x_n] \in R^{m\times n}$. The general objective function for learning the dictionary and classifier is defined as:

$$< D, W, A >= \underset{D,W,A}{\operatorname{argmin}} \|X - DA\|_F^2 + \eta \, \mathcal{L}\big(H, W, A\big) + \lambda_1 f(A) + \lambda_2 f(W) \tag{2.11}$$

where $D = [d_1, d_2, \ldots, d_k] \in R^{m\times k}$ is the dictionary, $A = [a_1, a_2, \ldots, a_n] \in R^{k\times n}$ denotes the sparse coefficients of input images, H is the label of training samples, W is the parameter of the classifier, $\mathcal{L}$ is the classification loss function, $f(A)$ and $f(W)$ are Lagrange constraints on the sparse coefficient matrix A and the classifier W, respectively and $\eta, \lambda_1, \lambda_2$ are scalars controlling the relative contributions of the corresponding terms.

In contrast, in the second group, each dictionary atom is predefined to correspond to a single class label, so that multiple sub-dictionaries are learned. 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. Mairal *et al.* [96] introduced a discriminative reconstruction penalty term in the KSVD model [2] for texture segmentation and scene analysis. Castrodad and Sapiro [18] learned a set of action-specific dictionaries with non-negative penalty on both dictionary atoms and representation coefficients. From the training images of each category, Wu *et al.* [143] learned active basis models for object detection and recognition. To encourage the dictionaries representing different classes to be as independent as possible, Ramirez *et al.* [113] introduced an incoherence promoting term to the dictionary learning model. Following [113], Wang *et al.* [132] presented a class-specific dictionary learning algorithm for sparse modeling in action recognition.

In the aforementioned methods, the representation residual associated with each class could be used for classification, but the representation coefficients are not enforced to be discriminative, so they are not used for classification. In the general model of class-specific dictionary learning, the atoms in the structured learned dictionary $D = [D_1, D_2, \ldots, D_K]$ have class label correspondences to the subject classes, where D_i is the sub-dictionary corresponding to class i. The sub-dictionary $D_i = [d_1, d_2, \ldots, d_{k_i}] \in R^{m \times k_i}$ is learned class by class as follows:

$$< D_i, A_i >= \underset{D_i, A_i}{\operatorname{argmin}} \sum_{i=1}^{K} \Big\{ \|X_i - D_i\, A_i\|_F^2 + \lambda_1 \|A_i\|_1 + \lambda_2 \mathcal{Q}(A_i) \Big\} \qquad (2.12)$$
$$s.t. \quad \|d_j\|_2 = 1 \quad \forall j$$

where $X_i \in R^{m \times n_i}$ contains all the training images from the i^{th} class, A_i is the representation of X_i over D_i, $\mathcal{Q}$ denotes the discrimination term for sparse representations, and λ_1, λ_2 are scalar parameters. Promising performance of class-specific dictionary representation have been reported in different recognition tasks [151], [149]; however, these dictionary learning methods might not be scalable to the problems with a large number of classes.

By considering discrimination from both reconstruction residual and coding vectors, Yang *et al.* [150] proposed a Fisher discrimination dictionary learning (FDDL) method, where the category-specific strategy is adopted for learning a structural dictionary, and the Fisher discrimination criterion is imposed on

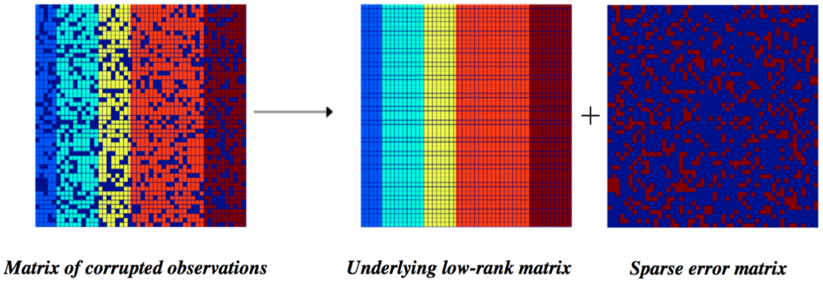


Figure 2.4: Overview of low-rank decomposition by RPCA [16]

the coding vectors to enhance class discrimination. As an extension, Yang *et al.* [149] introduced a latent dictionary learning method for sparse representation based image classification, which simultaneously learned a discriminative dictionary and a latent representation model based on the correlations between label information and dictionary atoms. In recent years, hybrid dictionary learning models have also been proposed to learn a shared dictionary and a set of class-specific dictionaries. Zhou *et al.* [165] learned a hybrid dictionary with a Fisher-like regularizer on the representation coefficients, while Kong *et al.* [73] learned a hybrid dictionary by introducing an incoherence penalty term to the class-specific sub-dictionaries. Instead of using a flat category structure, Shen *et al.* [122] proposed to learn a dictionary with a hierarchical category structure. Although the shared dictionary could make the learned hybrid dictionary more compact, balancing the shared and the class-specific parts is not a trivial task.

Some of the aforementioned dictionary learning methods perform well for classification and recognition tasks [150], [149], [14]; however, their performances dramatically deteriorate when the training data are contaminated heavily by occlusion, lighting and/or viewpoint variations and corruption.

2.5 Low-rank Approximation

In the recent years, low-rank matrix recovery, which efficiently removes noise from corrupted observations, has been successfully applied to a variety of computer vision applications, such as subspace clustering [87], background subtraction [120] and data compression [16]. Robust PCA [16], as a representative method showed impressive performance in background modeling and shadow removal. Given an observed and usually corrupted sample set X, RPCA decomposes X into a low-rank, clean sample set A and a sparse, noisy sample set E as follows:

$$\min_{A,E} rank(A) + \lambda \|E\|_0 \quad s.t. \quad X = A + E \tag{2.13}$$

where λ is a parameter that controls the weight of the noise matrix E. The minimization problem (2.13) is difficult to solve, since rank minimization and sparsity constraints are non-convex. We can replace the rank constraint by its surrogate, nuclear norm, and relax the l_0-norm and reformulate (2.13) as:

$$\min_{A,E} \|A\|_* + \lambda \|E\|_1 \quad s.t. \quad X = A + E \tag{2.14}$$

where $\|A\|_*$ is the sum of the singular values of A, and approximates the rank of matrix A. The overview of RPCA is shown in Figure 2.4. Several efficient solutions have been proposed for this problem such as Principal Component Pursuit (PCP) [16], robust subspace learning method [33] and augmented Lagrangian method (ALM) [86], among which ALM has attracted much more attention. The augmented Lagrangian form of (2.14) is:

$$L(A, E, Y, \mu) = \|A\|_* + \lambda \|E\|_1 + < Y, X - A - E > + \frac{\mu}{2} \|X - A - E\|_F^2 \tag{2.15}$$

where Y is the Lagrange multiplier, μ is a positive constant and $< K_1, K_2 >= tr(K_1^T K_2)$ is the inner product. Lin *et al.* [86] provided two ALM algorithms to solve Equation (2.15); namely exact and inexact ALM. Each iteration of the exact ALM involves solving sub-problem $(A_{k+1}^*, E_{k+1}^*) = \operatorname*{argmin}_{A,E} L(A, E, Y_k^*, \mu_k)$, and converges to the true solution in a small number of iterations. However, the algorithm can further speed up by using a fast continuation technique, thereby yielding the inexact ALM algorithm, which has been outlined in Algorithm 2.1. In this algorithm, SVD stands for singular value decomposition,

Algorithm 2.1 Inexact ALM Algorithm for Equation (2.15)

```
Input: Observation matrix $X, \lambda$.
Initialize: $Y$, $\mu > 0$, $\rho \geq 1$ .
while not converged do
    Solve $A_{k+1} = \underset{A}{\operatorname{argmin}}\ L(A, E_k, Y_k, \mu_k)$ as:
        $(U, S, V) = SVD(X - E_k + \mu_k^{-1} Y_k)$
        $A_{k+1} = U\, \Omega_{\mu_k^{-1}}[S]\, V^T$
    Solve $E_{k+1} = \underset{E}{\operatorname{argmin}}\ L(A_{k+1}, E, Y_k, \mu_k)$ as:
        $E_{k+1} = \Omega_{\lambda\mu_k^{-1}}\, [X - A_{k+1} + \mu_k^{-1} Y_k]$
    Update $Y$ as: $Y_{k+1} = Y_k + \mu_k\, (X - A_{k+1} - E_{k+1})$
    Update $\mu$ as: $\mu = \rho\mu$
    $k \leftarrow k + 1$
end while
Output: $(A_k, E_k)$
```

and the soft-thresholding (shrinkage) operator is defined as:

$$\Omega_\varepsilon[x] = \begin{cases} x - \varepsilon & if \quad x > \varepsilon \\ x + \varepsilon & if \quad x < -\varepsilon \\ 0 & otherwise \end{cases} \tag{2.16}$$

One major assumption in RPCA is that data are drawn from a single subspace. In practice, the underlying structure of data could belong to multiple subspaces. So, Liu *et al.* [87] developed low-rank representation (LRR) method for revealing the global structure of corrupted data drawn from multiple subspaces, as follows:

$$\min_{Z,E} \|Z\|_* + \lambda \|E\|_{2,1} \quad s.t. \quad X = DZ + E \tag{2.17}$$

where D is a dictionary that linearly spans the data space, Z is the lowest-rank representation of data X with respect to dictionary D and $\|E\|_{2,1} = \sum_{j=1}^{n} \left(\sum_{i=1}^{d} E_{ij}^2 \right)^{1/2}$. For subspace clustering, the observed data matrix itself is usually used as the dictionary *i.e.*, $D = X$ and in such cases, insufficient data sampling makes LRR ineffective. Also, the optimization of LRR requires multiple singular value decomposition (SVD) calculations that are very time consuming. In [89], LatLRR is proposed to solve the insufficient sampling problem by considering the effects of hidden data for representation. LRR and its variations achieved impressive results on subspace clustering and segmentation especially in noisy observations; however, this may not be efficient for finding a discriminative representation for classification task. Accordingly, some dictionary learning methods have been proposed by integrating rank minimization into sparse representation and have achieved impressive results, especially when the training data are contaminated heavily because of occlusion, lighting and viewpoint variations or corruption.

The combination of discriminative dictionary learning and low-rank approximation usually leads in better classification rate in both clean and noisy observations; but the advantage becomes more clear when the training and/or test samples are considerably noisy [82]. Generally speaking, these methods either use class-specific dictionary learning strategy and introduce low-rank constraint on sub-dictionaries for each class, or exploit shared dictionary learning technique, while considering a structured low-rank and sparse representation of coefficients. As a trendsetter, Chen *et al.* [24] employed low-rank matrix recovery to remove sparse noise from the training data class by class. A structural incoherence term is introduced to make the resulting low-rank dictionary for each class to be independent of each other. Ma *et al.* [95] integrated rank minimization into sparse representation by introducing low-rank constraint on sub-dictionaries for each class and achieved impressive face recognition results when corruption existed. To make the low-rank dictionary more discerning, Li *et al.* [82] proposed a discriminative dictionary learning with low-rank regularization ($\mathrm{D}^2\mathrm{L}^2\mathrm{R}^2$) for image classification. $\mathrm{D}^2\mathrm{L}^2\mathrm{R}^2$ adopts a class-specific dictionary learning strategy and imposes low-rank constraint on sub-dictionaries to make them robust to noise. The label information is explicitly incorporated through the Fisher discrimination function to make the learned dictionary more discriminative.

Unlike these class-specific methods, Zhang *et al.* [161] proposed a discriminative, structured low-rank method to explore the global structure among all training samples. A code regularization term with block-diagonal structure is incorporated to learn discriminative dictionary. It regularize the same class images to have the same representation. To capture the structure information globally in a more natural way, Li *et al.* [85] proposed a semi-supervised framework to learn robust face representation with classwise block-diagonal structure, which enhances intra-class similarities and inter-class differences. By exploiting the block-diagonal structure, a reconstructive and discriminative dictionary, and also discriminative representations are learned from images.

2.6 Dimensionality Reduction

In many areas of computer vision and pattern recognition, data are characterized by high-dimensional feature vectors, that are not only inefficient and computationally intensive, but the sheer number of dimensions often masks the discriminative signal embedded in the data [111]. So, for the efficient processing of a high-dimensional feature, its dimensionality has to be reduced without losing its intrinsic properties. In the literature, there are mainly two distinct ways for dimensionality reduction: (1) feature selection, that selects a subset of most representative or discriminative features from the input feature set, and (2) subspace learning (or feature transformation), that transforms the original input features to a lower dimensional subspace. The literature review of dimensionality reduction methods is beyond the scope of this thesis and here, we are just concerned about the techniques that have connections with sparse learning methods.

2.6.1 Subspace Learning

Subspace learning techniques transform dataset X with dimensionality D into a new dataset Y with dimensionality d ($d \ll D$), while retaining the geometry of the data as much as possible. The new, constructed feature space is usually a linear or non-linear combination of vectors from the original feature space. In general, neither the geometry of the data manifold, nor the intrinsic dimensionality d of the dataset X are known. Therefore, subspace learning is an ill-posed problem that can only be solved by assuming certain properties of the data [130].

In the last decade, a large number of linear and nonlinear subspace learning techniques have been proposed. Among them, some have been used in the literature to reduce the dimension of input data of sparse learning methods. The dimensionality reduction step significantly decreases the computation cost of sparse learning algorithms, and even makes data more effective due to ignoring irrelevant features. Wright *et al.* [140] used the linear projection of face images generated by a Gaussian random matrix for the initial dimensionality reduction, and called it Randomface. Since then, random projection (RP) [11] is often used for dimensionality reduction in the SRC and dictionary learning methods. RP transformation is independent of the training data set and it is extremely efficient to generate; however, this method does not take advantage of a priori label information for discriminative projection. Another common projection technique in sparse learning literature, is principal component analysis (PCA) [129], which constructs a low-dimensional representation of the data, that describes as much of the variance in the data as possible. This is done by finding a linear basis of reduced dimensionality for the data, in which the amount of variance in the data is maximal [130]. Locality preserving projections (LPP) [62] is another unsupervised dimensionality reduction technique, that aims to find a subspace that can preserve the neighborhood structure of the data. LPP and its extensions such as neighborhood preserving embedding (NPE) [61] have been developed based on the assumption that the data lie on a manifold which can be modeled by a nearest-neighborhood graph that preserves the local geomertic structure of the input space [130].

2.6.2 Feature Selection

The focus of feature selection is to select a subset of variables from the input data, which can efficiently describe the data, reduce effects of noise or irrelevant variables, and still provide good prediction results. In contrast to the subspace learning techniques which create new features, feature selection methods do not change the original representations of data variables, and this is preferred when we are required to keep the original physical meanings of each feature.

Based on different strategies of searching, feature selection can be classified into three methods including filter, wrapper and embedded methods [58]. Relying on the characteristics of data, filter models evaluate features without utilizing any classification or clustering algorithms. A typical filter algorithm consists of two steps. First, all features are ranked according to certain criteria. Feature evaluation could be either univariate or multivariate. In the univariate scheme, each feature is ranked independently of the feature space, while the multivariate scheme evaluates features in an batch way [58]. Then, the features with highest rankings are chosen to induce classification models. In the past decade, a number of performance criteria have been proposed for filter-based feature selection including mutual information (MI) [76], [47], [155], [110], minimal redundancy maximal relevance (mRMR) [110], Laplacian score (LS) [60], Fisher score (FS) [38] and reliefF [70]. Wrapper methods utilize the intended learning algorithm itself to evaluate the quality of selected features. By changing the subset generation technique and subset evaluation measure, a different wrapper algorithm can be generated. A wide range of search strategies including hill-climbing [72], best-first, branch-and-bound [100], and genetic algorithms [55] are used in these methods. Then, they run the classifier many times to assess the quality of selected subsets of features to find the optimal subset. This step is computationally expensive, but provides better predictive accuracy estimates compared to filter methods. Finally, embedded models perform feature selection via model construction. By incorporating the feature selection as a part of the training process, embedded models have the advantages of wrapper and filter methods, since they include interactions with the classifier, and are far less computationally intensive than wrapper methods.

In the recent years, the embedded model is gaining increasing interest in feature selection research due to its superior performance. Several embedded feature selection algorithms applied l_0-norm [137], [65] and l_1-norm [90], [166], [164] constraints to existing learning models in order to achieve a sparse solution. For instance, to obtain a sparse decision rule for

SVM, Bradley *et al.* [12] and Zhu *et al.* [166] proposed l_1-sparse SVM, which uses l_1-norm as the regularizer to perform feature selection. Ng *et al.* [102] used logistic regression with l_1-norm regularization for feature selection. By combining l_1-norm and l_2-norm, a more structured regularization called Hybrid Huberized SVM [134] was proposed. [90] developed a model with $l_{2,1}$-norm regularization to select features shared by multi tasks. Nie *et al.* [104] also employed joint $l_{2,1}$-norm minimization on both the loss function and the regularization to select the most relevant features.

A widely accepted criterion is to select features that best preserve the manifold structure of the data. Yang *et al.* [153] explored this idea by combining the manifold learning and $l_{2,1}$-norm minimization into joint feature selection, and proposed an unsupervised feature selection algorithm in batch mode. As an alternative to exploiting discriminative information, Hou *et al.* [63] proposed a non-negative discriminative feature selection method which performs spectral clustering and feature selection simultaneously in a joint optimization framework. To make the selected feature more discriminative, Yan *et al.* [144] developed sparsity preserving score to evaluate the importance of features. The objective function jointly selects features by projecting the original high-dimensional data to a low-dimensional space through a special binary projection matrix. This was further improved in [145] by introducing the SRC measurement criterion into feature selection, and designing a joint sparse discriminative feature selection method. Considering the decision rule of SRC, their objective function aims to find a subset of features whose components could be well approximated by the linear combination of other components in the same class. This is achieved by minimizing the ratio of intra-class to inter-class reconstruction residual in the subset of selected features.

2.7 Joint Dimensionality Reduction and Dictionary Learning

As previously mentioned, to deal with high-dimensional feature vectors in dictionary learning process, dimensionality reduction is performed first on the training samples using either PCA or RP, and then the dimensionality reduced data are used as the input data for learning the dictioanry and sparse coefficients. However, recent studies revealed that the pre-learned dimensionality reduction matrix neither fully promotes the underlying sparse structure of data [103], [158], nor preserves the best features for dictionary learning [44]. Intuitively, the dimensionality reduction and dictionary learning processes should be jointly conducted for a more effective classification.

Only a few works have discussed the idea of jointly learning the projection of training samples and dictionary, and mostly reported more competitive performance than conventional dictionary learning methods. [158] presented a simultaneous projection and ditionaey learning method using a carefully designed sigmoid reconstruction error. The data is projected to an orthogonal space where the intra- and inter-class reconstruction errors are minimized and maximized, respectively for making the projected space discriminative. However, [53] showed that the dictionary learned in the projected space is not more discriminative than the one learned in the original space. JDDLDR method [44] jointly learned a dimensionality reduction matrix and a discriminative dictionary, and achieved promising results for face recognition. The discrimination is enforced by a Fisher-like constraint on the coding coefficients, but the projection matrix is learned without any discrimination constraints. Nguyen *et al.* [103] proposed a joint dimensionality reduction and sparse learning framework by emphasizing on preserving the sparse structure of data. The so-called sparse embedding (SE) method, can be extended to a non-linear version via kernel tricks that leads to a better classification accuracy. SE outperforms convetional dictionary learning methods especially in small-sized datasets; however, it fails to consider the discrimination power among separately learned, class-specific dictionaries, such that it is not guaranteed to produce improved classification performance [74].

Ptucha *et al.* [111] integrated manifold-based dimensionality reduction and sparse representation within a single framework, and presented a variant of the KSVD algorithm by exploiting a linear extension of graph embedding (LGE). The LGE concept is further leveraged to modify the KSVD algorithm for cooptimizing a small, yet overcomplete dictionary, a projection matrix and the coefficients. Yang *et al.* [147] simultaneously learned a dimensionality reduction matrix and a set of class-specific sub-dictionaries, and alos utilized both representation residuals and coefficients for the classification purpose. Most recently, Liu *et al.* [91] proposed a joint non-negative projection and dictionary learning method. The discrimination is achieved by imposing graph constraints on both projection and coding coefficients that maximizes intra-class compactness and inter-class separability. Although, great successes have been reported by some of the aforementioned joint dimensionality reduction and dictionary learning methods in different classification and recognition tasks, most of these methods cannot handle noisy observation such as occluded and corrupted data, and large intra-class variations, which is very common in real-world datasets.

Chapter 3

Crowd Counting using Sparsity

3.1 Introduction

Counting pedestrians in videos is a topic of significant interest in areas such as safety and security, resource management, urban planning and visual surveillance systems. Crowd counting could be a challenging task due to severe occlusions, scene perspective distortions and diverse crowd distributions. The literature on the pedestrian counting includes three conceptually different techniques: counting by detection, counting by clustering and counting by regression.

In counting by detection [35], [79], a classifier is trained to learn a model for a single person, based on some features such as histogram of oriented gradients (HOG) [31]. In these methods, we must provide the system with a large set of object examples, properly labeled and localized, that represent most of the possible views and appearances of the object. A trained classifier is then applied in a sliding window fashion across the test image to detect pedestrian candidates. The detection performance can be further improved by adopting part-based detection techniques, especially head-shoulder detectors [83], or tracking the detected objects over time [163]. However, as the crowd becomes larger and denser, detection and tracking tasks become impractical due to the small scale of individuals and occlusions. In contrast, counting by clustering [13], [112] is based on identifying and tracking visual features over time. These techniques assume that the crowd is composed of individual entities, each of which has a unique yet coherent motion pattern that can be clustered to estimate the number of people. However, coherently moving features usually do not belong to the same person, especially in crowded environments and these methods need sophisticated trajectory management. Counting by regression [21], [19], [25] methods estimate the number of people in an image by learning a direct mapping from low-level image features to the count, without intermediate steps. In these methods, a region of interest (ROI) is detected and several low-level features with complementary nature are extracted from crowd segments, and then a regression function is trained to predict the count. The extracted features usually include foreground segment features (*e.g.*, area, perimeter, perimeter-area ratio, perimeter edge orientation and blob count), edge features (*e.g.*, number of edge pixels, edge orientation), and texture and gradient features (*e.g.*, gray level co-occurrence matrix and its derived features such as homogeneity, entropy, etc.). Popular regression models that are used for estimation consists of linear function [81], ridge regression [25], support vector regression [142] and Gaussian process regression [21], [19]. By a different viewpoint, Lempitsky *et al.* [80] introduced an object counting method through pixel-level object density map regression. Following [80], Fiaschi *et al.* [46] used a random forest to regress the object density and improved training efficiency. Loy *et al.* [23] also introduced semi-supervised regression and data transferring methods to reduce the amount of training data needed for training of regression methods.

Although regression-based methods are feasible for crowded environments, and could achieve promising results in these scenes, they still have important disadvantages. Loy *et al.* [93] performed extensive experiments on a wide range of regression-based methods using various benchmark datasets, and revealed the following facts: (1) The performance of these methods is dependant on the selected feature set, to a large extent. Also, the optimal feature set would be different in various crowd structures and densities, because the number of features carried by one pedestrian is heavily affected by camera perspective and crowd density. (2) The actual performance of a regression model can be quite different from what one may anticipate, subject to the nature of data, especially when it is applied to unseen crowd density. Besides, most of these regression models suffer from poor extrapolation beyond the training data range. (3) Given an unseen scene, the model has to be trained from scratch and evaluated to find the optimal feature set and the regression model. Considering these challenges, we propose two crowd counting methods based on compressed sensing and sparse representation theories, each of which is capable of resolving some of the aforementioned issues. We discuss them in the following sections.

It should be underlined that recently deep neural networks and especially convolutional neural network (CNN) have been widely applied to many computer vision applications such as classification. Generally, the CNN architecture consists of multiple trainable stages stacked on top of each other, followed by a supervised classifier. Sermanet *et al.* [119] showed that the features extracted from CNNs are more effective than hand-crafted features for many applications, and these features can be reused in other tasks than classification. Accordingly, some researchers have recently adopted CNN as the basic framework to learn efficient features for crowd counting. For instance, [131] trained a deep network to predict the total crowd count in an image patch. In [157], a CNN is trained alternatively with two related learning objectives, crowd density and crowd count with dual-loss functions. Segui *et al.* [118] introduced the problem of object representation as an indirect learning problem, cast as a learning to count approach using CNN. It should be pointed out that all of these deep learning-based methods have been released after our methods have been proposed and published [48], [49].

3.2 Crowd Counting using Compressed Sensing

In this section, we propose a simple counting method based on the integration of image retrieval framework, and compresses sensing (CS) theory. We assume that we have a large amount of labeled training data of a pedestrian dataset. For this task, each image is labeled with the number of people present in the image. In a generic image retrieval framework, some features are extracted

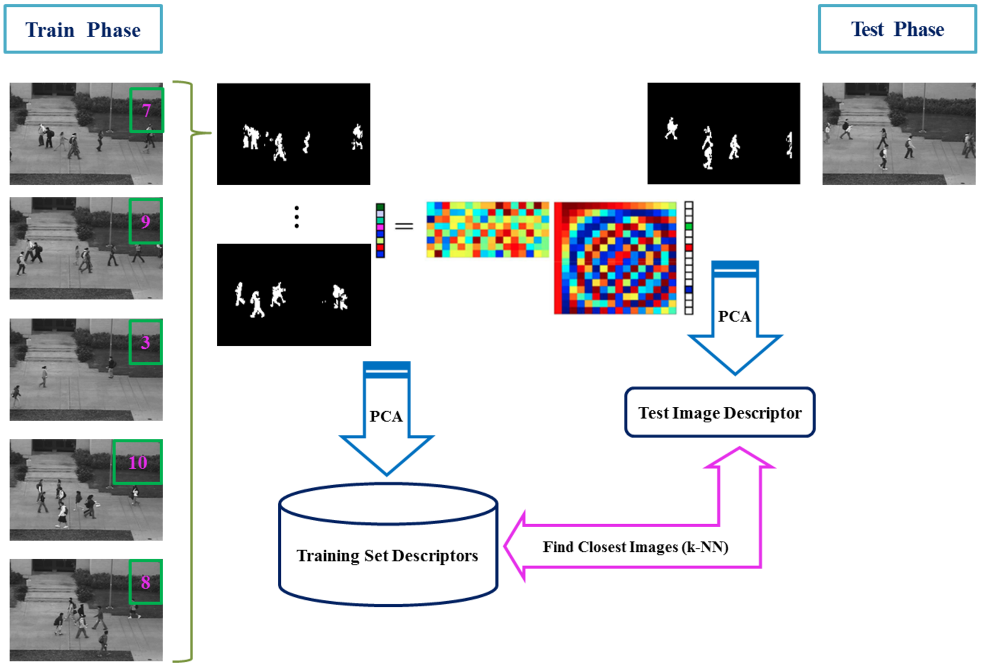


Figure 3.1: Overview of the proposed RCS-Count method

from each image and during the test phase, the closest images to the test image are retrieved according to a similarity measure. So, if we have a large enough pedestrian dataset that can span the variation under testing conditions, the pedestrian count for a query image can be efficiently estimated by retrieving few closest images from the training set, and computing the average or majority of the counts associated with the retrieved images. For the feature extraction part, we use some global image descriptors that characterize an entire image with a single vector. To further improve the accuracy of this retrieval-based framework, we also introduce a global image descriptor based on compressed sensing theory, and utilize it to estimate the count. Figure 3.1 shows the overview of the proposed *Retrieval-based Counting using Compressed Sensing* (RCS-Count) method.

3.2.1 CS-based Image Representation

We aim to use CS to represent visual data and propose a new image representation scheme for people counting. In essence, CS exploits prior knowledge about the sparsity of a signal x in a linear transformation domain, in order to develop efficient sampling and reconstruction. Let $x \in R^N$ be a signal that can be sparsely represented in a certain domain by the transformation matrix $\Psi \in R^{N \times N}$ as $x = \Psi s$. The assumption of sparsity means that only $K(K << N)$ non-zero coefficients in s are sufficient to represent x exactly. We do not observe the K-sparse signal x directly, instead we record $M << N$ non-adaptive linear measurements as $y = \Phi x$, where $\Phi \in R^{M \times N}$ is a measurement matrix made up of random orthobasis vectors. As discussed in Section 2.1, CS theory states that we can reconstruct x (or, equivalently s) accurately from y if Φ and Ψ are incoherent and also Φ satisfies the restricted isometry property. In this case, the recovery works with high probability if M is in the order of $Klog(N)$, using the following optimization:

$$\hat{s} = \underset{s}{\operatorname{argmin}} \|s\|_1 \quad s.t. \quad y = \Phi \Psi s \tag{3.1}$$

When CS is applied into practical applications such as image representation, there are still some issues to be considered. Natural images are not generally sparse, but compressible in certain domains such as Fourier or Wavelet. The goal of this work is to count the number of people in a scene, and people are mainly the moving objects in a scene. So, separating out foreground objects from the background, which is called background subtraction, gives us the most natural sparse representation of an image for our application. As the first step, we utilize an adaptive Gaussian mixture model [167] to perform background subtraction to obtain the sparse representations of images. It has been shown that random Gaussian and partial Fourier matrices satisfy the RIP with high probability [116]. In order to take the random measurements, here we use Fourier transform, to also benefit from its translation invariant property. The measurement vector y is created according to (3.1), and the image descriptor is then built using the magnitude of these random measurements. Since $M << N$, using the CS-based descriptor leads to significant compression in a generic retrieval-based framework; however, we also PCA to further reduce the dimension of the descriptor and the computational complexity. So,

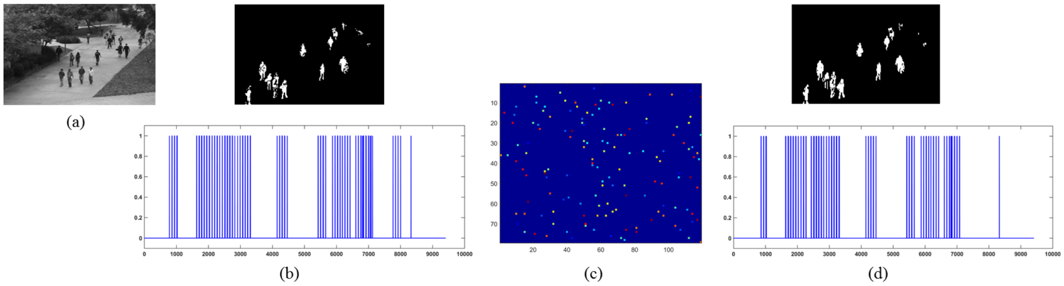


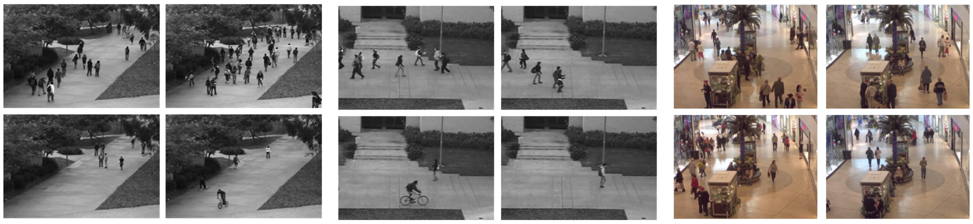
Figure 3.2: Recovery of a signal in the proposed RCS-Count method on the UCSD dataset (a) Original image (b) Binary image obtained from background subtraction and its vectorized form (c) CS measurements in Fourier domain (d) Recovered binary image and its vectorized representation

the final CS-based descriptor is built as follows:

$$\widetilde{y} = \mathcal{A}\,|y| \tag{3.2}$$

where $\mathcal{A} \in R^{d \times M}$ is the projection matrix of PCA, and $|y|$ denotes the magnitude of random measurements in Fourier domain. This, gives us a d-dimensional feature vector, which clearly $d << N$.

Figure 3.2 gives an example to explain the signal recovery in our method. Figure 3.2a shows a down-sampled original image from the UCSD dataset [21] with the size of $N = 119 \times 79 = 9,401$. The result of background subtraction, and the vectorized binary image are illustrated in Figure 3.2b. This representation is very sparse with only $K = 320$ non-zero elements. Figure 3.2c is the CS measurements in Fourier domain with only $M = 340$ measurements. Figure 3.2d is the recovered image from the CS measurements, through l_1-minimization algorithm, and its vectorized form. Clearly, the signal has been well recovered. The proposed descriptor represents a $9,400$-dimensional sample image with just a 340-dimensional vector, and it can even be reduced further to $d = 34$ after applying PCA. Benefiting from the produced compact code, fast image search can be carried out, which dramatically reduces the computational cost, and also optimizes the efficiency of the search.



(a) UCSD-Peds1 (b) UCSD-Peds2 (c) Mall

Figure 3.3: Sample images from benchmark crowd counting datasets

3.2.2 Experiments

Datasets: We carry out experiments on two benchmark pedestrian datasets.

- **UCSD Dataset**: The UCSD dataset [21] is the largest benchmark pedestrian dataset in terms of number of frames, which was collected from two viewpoints overlooking a pedestrian walkway in the UCSD campus. Peds1 and Peds2 sequences have oblique and side views with around $33,000$ and $34,000$ frames, respectively. Peds1 is generally more crowded with $(0 \sim 46)$ pedestrian count (*i.e.,* the minimum and maximum number of people in the provided ROI), compared to Peds2 with the count of $(0 \sim 15)$. Also, Peds1 is substantially more challenging than Peds2 because of the oblique view, travelling bicycles, skateboarders and golf carts. Example frames from these datasets are shown in Figures 3.3a and 3.3b, respectively. The videos were recorded at 10 fps with a frame size of 238×158. The first $4,000$ frames of each video sequence were originally used for ground-truth (GT) annotation; however, since our model should be validated on a large dataset, we exhaustively annotated the whole dataset on the provided ROIs for our experiments.

- **Mall Dataset**: The shopping mall dataset [93] was collected from a publicly accessible webcam during peak hours in a mall. This dataset covers crowd densities from sparse to crowded $(13 \sim 53)$, as well as diverse activity patterns (static and moving crowds), under large range of illumination conditions at different times of the day. A total number of $2,000$ images were captured at $1 \sim 2$ fps, with a frame size of 320×240. Some sample images from this

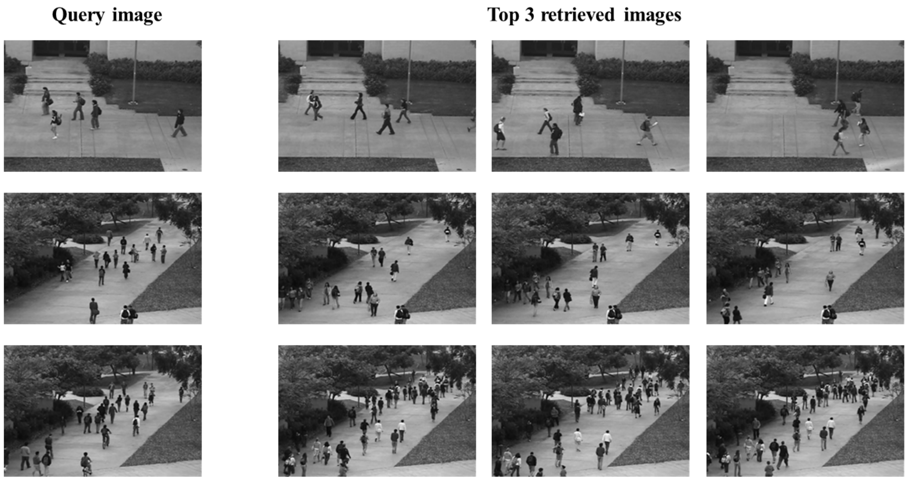


Figure 3.4: Top three retrieved images from the UCSD dataset by RCS-Count method

dataset are illustrated in Figure 3.3c.

Image Descriptors: For each dataset, a ROI has already defined to exclude the non-corridor/non-pathway regions in the scene, and features are extracted from this ROI. For CS-based descriptor, we use the compression rate 83% $i.e., N = 6M$, and then apply PCA on M measurements with over 90% reduction in the size of descriptor. Specifically, for the UCSD dataset we have $N = 37,604; M = 6,267; d = 626$ and for the Mall dataset the numbers would be as $N = 76,800; M = 12,800; d = 1,280$.

In addition to the proposed CS-based descriptor, we would evaluate the performance of some popular global image descriptors in this framework. Global descriptors do not require any keypoint detection, and represent an image with a single vector, which means they are fast to build and efficient to store for retrieval-based methods. As the simplest global descriptor, we can use the down-sampled image itself; however, it sounds a naive option. In the experiments we used half of the original size of images. Gist [106] represents an image in terms of its response to a bank of Gabor filters. An image is divided into small tiles, and the final feature descriptor is the mean response of these tiles to steerable filters at different scales and orientations. Specifically, the image is first decomposed by a bank of multi-scale oriented filters, which is tuned to 8 orientations and 4 scales. Then, the output magnitude of each filter is averaged over 16 non-overlapping windows arranged on a 4 × 4 grid. The resulting image representation is a $4 \times 8 \times 16 = 512$-dimensional vector.

HOG [31] counts the occurrences of gradient orientations in localized portions of an image. An image is divided into small cells and for each cell a histogram of gradient directions is computed and discretized into angular bins. Groups of adjacent cells are considered as blocks in order to normalize these histograms, and a set of these block histograms represents the descriptor. In our experiments, we use the following settings for extracting HOG descriptor; cell-size: 20 × 20, block-size: 2 × 2, number of overlapping cells between adjacent blocks: 1 × 1, and number of orientation histogram bins: 9. This leads to $2,160$ and $5,940$-dimensional feature vectors for the UCSD and Mall datasets, respectively. We also utilize whole image SIFT (WI-SIFT) and whole image SURF (WI-SURF) descriptors, which are the global versions of local descriptors SIFT [92] and SURF [9], respectively. In this case, the center of an image is considered as the only detected keypoint and the image descriptor, describes its neighborhood including whole image. For WI-SIFT and WI-SURF descriptors, a set of orientation histograms is created on 4 × 4 pixel neighborhoods with 8 bins each, around the only keypoint of image located at the centre. So, this is leading to a 128-dimensional descriptor for WI-SIFT and WI-SURF features vectors.

Evaluation metrics: In order to evaluate the accuracy of estimations, we employ three widely used metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Mean Deviation Error (MDE). These errors are defined as:

$$MAE = \frac{1}{N_t}\sum_{i=1}^{N_t}|y_i - \hat{y}_i| \ , \ MSE = \frac{1}{N_t}\sum_{i=1}^{N_t}(y_i - \hat{y}_i)^2 \ , \ MDE = \frac{1}{N_t}\sum_{i=1}^{N_t}\frac{|y_i - \hat{y}_i|}{y_i} \tag{3.3}$$

where N_t is the number of test images, and y_i, $\hat{y}_i$ are the true and predicted counts in the i^{th} test image, respectively. It is worth pointing out that in contrast to the other two metrics the MDE is more indicative, as it takes the

Table 3.1: The error rates of different methods on the UCSD-Peds1 dataset

Method	MAE				MDE			
	5K	15K	20K	25K	5K	15K	20K	25K
RCS-Count	2.92	1.70	1.62	1.37	0.19	0.11	0.10	0.08
Retrieval-Gist	4.21	2.62	2.17	1.76	0.28	0.15	0.14	0.11
Retrieval-HOG	3.71	1.81	1.63	1.47	0.22	0.13	0.10	0.09
Retrieval-WI-SIFT	5.38	2.32	2.13	1.86	0.33	0.26	0.23	0.19
Retrieval-WI-SURF	5.98	2.96	2.90	2.40	0.36	0.30	0.25	0.21
Retrieval-Sub-sample	8.65	4.50	4.28	3.10	0.47	0.36	0.32	0.26
KRR [6]	4.02	7.39	8.10	9.11	0.22	0.29	0.29	0.30
SVR [142]	4.44	6.87	6.00	5.10	0.23	0.25	0.25	0.24
GPR [21]	4.43	6.98	5.89	6.69	0.25	0.28	0.26	0.28

level of crowdedness of the i^{th} frame into account.

In all the following experiments, nearest neighbor classifier is used to retrieve top three images based on the Euclidean distance of descriptors, and the majority of retrieved counts is considered as the predicted count of the query image. All results are averaged over 10 trials and mean errors are reported. Figure 3.4 demonstrates some sample test images from the UCSD dataset belonging to low, medium and high density scenes, as well as top three retrieved images from the depository using CS-based descriptor and $15,000$ training images. It is observed that query image and retrieved images have roughly the same number of people.

First, to evaluate the effectiveness of exploiting more training data in the retrieval-based framework, we change the size of training data in the UCSD dataset. We randomly choose $\{5,000; 15,000; 20,000; 25,000\}$ training samples in each of the Peds1 and Peds2 datasets, and use the rest for the testing. We compare our CS-based descriptor with aforementioned global descriptors. In addition, we compare our method with regression-based counting methods. For these methods, three categories of features including segments, internal edges and texture are extracted from images, and then concatenated

Table 3.2: The error rates of different methods on the UCSD-Peds2 dataset

Method	MAE				MDE			
	5K	15K	20K	25K	5K	15K	20K	25K
RCS-Count	0.74	0.34	0.28	0.17	0.13	0.11	0.10	0.07
Retrieval-Gist	0.89	0.38	0.31	0.16	0.16	0.13	0.11	0.09
Retrieval-HOG	0.85	0.36	0.32	0.19	0.17	0.13	0.11	0.09
Retrieval-WI-SIFT	0.91	0.37	0.32	0.19	0.20	0.14	0.13	0.10
Retrieval-WI-SURF	1.10	0.47	0.42	0.24	0.21	0.15	0.15	0.11
Retrieval-Sub-sample	1.93	0.69	0.59	0.31	0.27	0.17	0.14	0.12
KRR [6]	1.82	3.00	3.18	3.37	0.23	0.26	0.28	0.30
SVR [142]	2.10	2.76	2.00	1.86	0.20	0.23	0.20	0.19
GPR [21]	1.61	2.46	2.12	1.90	0.18	0.22	0.20	0.16

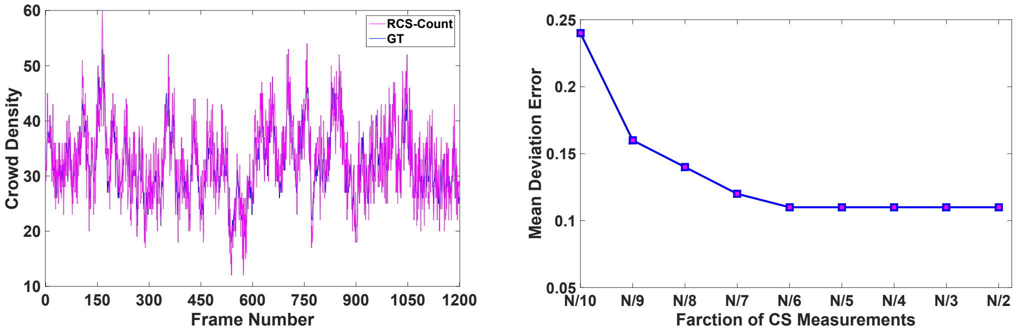
features are fed to the kernel ridge regression (KRR) [6], support vector regression (SVR) [142] and Gaussian processes regression (GPR) [21] models. The error rates on the Peds1 and Peds2 datasets are summarized in Table 3.1 and Table 3.2, respectively. As expected, we have higher errors for the Peds1 dataset because of larger crowd, moving bicycles, skate-boarders, and golf carts. The results indicate when there is a large amount of training data *e.g.*, $25,000$ images, retrieval-based method using either of descriptors, outperforms the regression-based methods. As the number of training samples grows, both MAE and MDE are decreased for all of the evaluated descriptors, which proves the effectiveness of this framework for large-scale datasets. Although, in smaller training sets *e.g.*, $5,000$ images, the proposed method still achieves better performance than the regression-based methods, the difference is not significant. Unlike our method, involving more data is not always helpful for regression-based methods. By enlarging the training set, the degree of non-linearity is increased in the chosen feature space, and the regression function might not be able to capture it well. This, may justify the higher error rates of these methods in larger datasets. These results also reflect that a simple ridge regression model achieves comparable or better performance compared to the more complex Gaussian processes regression model.

Table 3.3: The error rates of different methods on the Mall dataset

Method	MAE	MSE	MDE
RCS-Count	3.40	16.10	0.09
Retrieval-Gist	3.70	19.40	0.12
Retrieval-HOG	3.43	17.76	0.11
Retrieval-WI-SIFT	3.90	19.10	0.16
Retrieval-WI-SURF	3.98	19.87	0.18
Retrieval-Sub-sample	4.12	21.69	0.39
KRR [6]	3.52	17.20	0.11
SVR [142]	3.51	17.18	0.10
GPR [21]	3.72	19.10	0.12

For the Mall dataset, following the settings in [25], we use the first 800 frames for training and keep the remaining $1,200$ frames for testing, and show the results in Table 3.3. We observe that our method yields competitive results to regression-based methods in this small and challenging dataset. Our method gains from more data, and boosts the accuracy of crowd counting much more in large datasets. Also, in contrast to the regression models, in which different features can be more useful given different crowd configurations and densities, the proposed CS-based descriptor shows superior performance in all the evaluated scenarios. Figure 3.5a shows the crowd counting estimations and the ground-truth for $1,200$ testing images of the Mall dataset. One may say the estimates track the ground-truth well in most cases.

Finally, we review the effect of number of CS measurements (M) on the crowd counting performance. In this experiment, we randomly select $15,000$ training images from the Peds1 dataset and use the rest for testing, and then adopt CS-based descriptor to represent images. Figure 3.5b illustrates the MDE on the test set, and the horizontal axis shows the ratio of M/N. Even if the number of measurements is a small portion of the image size, the proposed RCS-Count method obtains small error rates. Also, MDE is decreased significantly when we add a few more measurements.



(a) (b)

Figure 3.5: The performance of RCS-Count method (a) Estimated count versus ground-truth on the Mall dataset (b) MDE as a function of number of CS measurements on the UCSD-Peds1 dataset

3.2.3 Summary

In this section we proposed a crowd counting method based on image retrieval, which uses an image descriptor to estimate the count. In addition to reviewing the performance of prevailing global descriptors, we introduced a compact global image descriptor based on CS theory. The experimental results reveal that proposed RCS-count method performs well to estimate crowd density in comparison with state-of-the-art regression-based methods. The advantage of this method is particularly significant, when the pedestrian dataset is large.

3.3 Crowd Counting using Sparse Representation

In this section, we propose a crowd counting method based on the integration of sparse representation-based classification (SRC), random projection and semi-supervised elastic net. Figure 3.6 illustrates the proposed method, known as SRP-Count. Like before, we assume that we have a large amount of labeled training data of a pedestrian dataset, in which each image has been annotated with the number of people present in it. We treat the counting task as a classification problem, where the pedestrian count is considered as the class, and images with the same number of objects are considered as the same class.

Suppose that we have K distinct classes of counts, and let $X =$

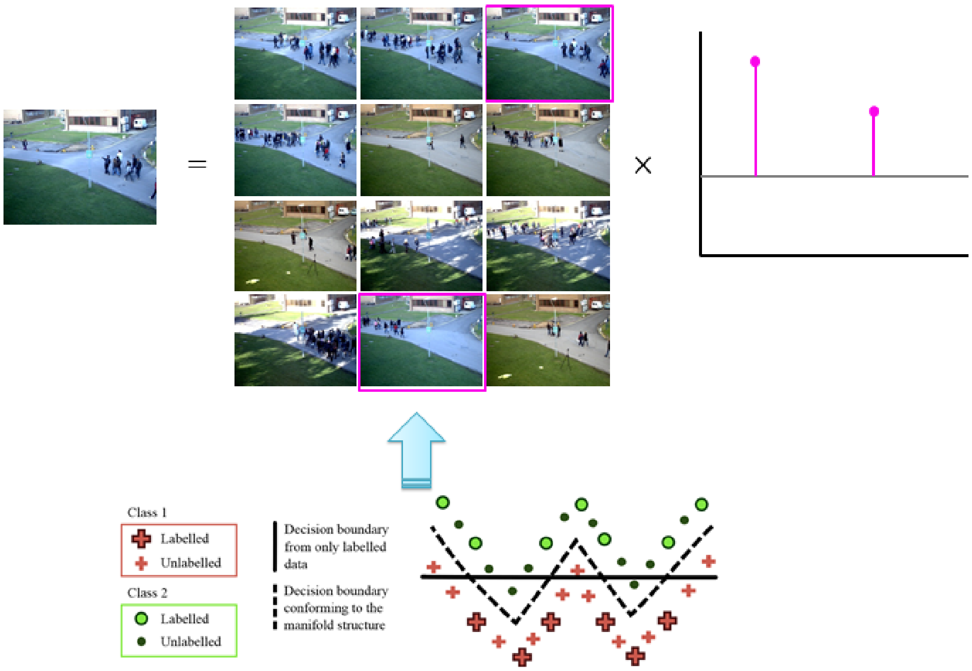

=
×
Class 1
Labelled
Unlabelled
Class 2
Labelled
Unlabelled
Decision boundary from only labelled data
Decision boundary conforming to the manifold structure

Figure 3.6: The overview of proposed SRP-Count method

$[X_1, X_2, \ldots X_K] \in R^{m \times N}$ be the set of training samples, where X_i is the subset of the training samples from class i, m is the data dimension and N is the total number of training data. According to the sparse representation theory, given sufficient training samples from the i^{th} class, any query image x_{ts} from the same class, will approximately lie in the linear span of the training samples associated with class i. Since the class label of the test image is initially unknown, the linear representation of x_{ts} is considered in terms of all training samples as $x_{ts} = X\alpha$. Here, α is the coefficient vector whose entries are ideally all zero, except those associated with the i^{th} class, such that the dominant non-zero coefficients can reveal the true class of test image. As discussed in Section 2.2, the sparsest solution to this problem is found by the following l$_1$-minimization problem:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \, \|x_{ts} - X\alpha\|_2^2 + \lambda \|\alpha\|_1 \tag{3.4}$$

The naive dictionary X can be built using raw images or some extracted features from them. Our experiments on the proposed RCS-Count method in Subsection 3.2.2 suggest that global image descriptors, and especially Gist [106] and HOG [31] are generally effective in describing the crowd density. Furthermore, the learnt features from deep neural networks and especially CNNs, have recently shown great potential in various vision tasks, and outperfromed many of the hand-crafted features. These features have demonestrated excellent performance not only on the ImageNet [34] classification task the CNN was originally trained for, but also on a variety of other recognition tasks [36], [119]. So, to benefit from the deep features in our framework, we take a CNN pre-trained on the ImageNet dataset, remove the last fully-connected layer and then treat the rest of the CNN as a fixed feature extractor for the pedestrian dataset. For instance, in the popular AlexNet architecture [75] we use the output of the last layer (the layer before the classification layer), known as fc7 as the generic image descriptor, which generates a 4096-dimensional vector for every image.

Both hand-crafted and deep features are somewhat high-dimensional. To reduce the computational cost, while retaining as much of the information content of the data as possible, we may use a subspace learning technique *e.g.*, PCA, and project the feature vectors into a lower-dimensional feature space $R^d (d << m)$. The fact is that SRC usually needs a large training set to span all the testing variations, and PCA with computational complexity of $O(m^2 N + m^3)$ could be computationally expensive for our application [11]. To address this problem we exploit random projection (RP) [32], which projects data into a low-dimensional subspace randomly such that the discriminative information can be approximately retained. As shown in [11], RP offers clear benefits over PCA: (1) Compared to PCA, RP is much simpler and less expensive with computational complexity of $O(dmn)$. (2) As the projected dimension is decreased and drops below a threshold, RP faces a gradual degradation in the performance; however, PCA extremely distorts the data in smaller dimensions, and this is mainly because the performance of PCA is dependent on the sum of omitted eigenvalues. (3) Another disadvantage of PCA is that it preserves larger distances better than the smaller ones. This is mainly because PCA aims at preserving the matrix of pairwise scalar products, in the sense that the sum of squared differences between the original and reconstructed scalar products should be minimal.

The key idea of random projection arises from the Johnson-Lindenstrauss lemma [69], that states if points in a vector space are projected onto a randomly selected subspace of suitably high dimension, the distances between the points are approximately preserved. Based on this principle, Gaussian random matrices and a few sparse random matrices with $\{0, \pm 1\}$ elements have been proposed for random projection, which the former yields stronger distance preservation [94]. In Gaussian random matrix, the entries are independently sampled from a zero mean normal distribution, and each row is normalized to unit length. Using the random matrix $\Phi \in R^{d \times m}$, the Equation (3.4) is converted to:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|\Phi x_{ts} - \Phi X \alpha\|_2^2 + \lambda \|\alpha\|_1 \tag{3.5}$$

As explained in Section 2.2, there are some efficient and fast solvers for the l_1-minimization (3.5), such as Homotopy [37] and ALM [146]. The principles of ALM can also be applied on the dual problem of (3.5), and according to comprehensive comparisons provided in [146], the dual ALM (DALM) performs the best compared to other l_1-minimizers in recognition experiments, and also scales well in terms of the number of classes, with computational complexity of $O(m^2 + mN)$. If the solution is very sparse with s non-zero coefficients, we can also exploit Homotopy with complexity of $O(sm^2 + smn)$. It is important to realize that the combination of random projection and fast l_1-minimizers provide a viable solution to real-time crowd counting, even in large-scale datasets.

Once the sparse vector $\hat{\alpha}$ is recovered by (3.5), we estimate the count for unseen test images. We classify query image x_{ts} based on how well the coefficients associated with the i^{th} class can reproduce it, as follows:

$$label(x_{ts}) = \underset{i}{\operatorname{argmin}} \|\Phi x_{ts} - \Phi X \delta_i(\hat{\alpha})\|_2 \tag{3.6}$$

where $\delta_i(\hat{\alpha})$ is a function that selects the coefficients associated with the i^{th} class.

3.3.1 Under-sampled SRC

It is commonly believed that SRC requires a rich set of training images of every class that can span the variations under testing conditions; however, in real-world it is relatively rare to have a labeled pedestrian dataset of sufficient size. To avoid the tedious and laborious task of manual image annotation, here we utilize the abundant unlabeled data that can be collected easily. Specifically, in pedestrian datasets there is a large number of unlabeled images which provide useful topological information. We present a semi-supervised elastic net (SSEN) model to estimate count for these unlabeled images, by utilizing the sequential information amongst them. To provide a large and diverse training set for SRC, we iterate count estimation process by SSEN, and each time the most confident recently labeld samples and their predicted labels, are added to the initial training set. We elaborate these steps as follows.

Elastic net (EN) is a shrinkage and selection method for producing a sparse model with good prediction accuracy. Given a set of l training samples and their labels as $\{(x_i, y_i)\}_{i=1}^{l}$, EN optimizes the following function:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \|y - \mathcal{X}\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \tag{3.7}$$

where $\mathcal{X} \in R^{N\times m}$ is a row-stacked matrix of $x_i s$, vector $y \in R^{N\times 1}$ consists of their labels y_i, and λ_1, λ_2 are scalar parameters.

One common way to construct a semi-supervised algorithm is to add unlabeled data as a regularization term, and in our application this term is created by exploiting the sequential information among unlabeled frames [127]. Specifically, if we have a time-sequenced image (frame) set $X = \{x_1, x_2, \ldots, x_N\}$, the pedestrian quantities change slightly or even remains the same in every p sequential frames. We assume the training dataset is the union of several disjoint sets as $X = S_1 \cup S_2 \cup \cdots \cup S_r$, where $S_i = \{x_{i,1}, x_{i,2} \ldots x_{i,p}\}$. The window width p can be set to a constant value, or even vary as the scene changes, using for instance a simple background subtraction method. Once p is determined, we build a neighboring frame set Ω by all images pairs belonging to all sets. Thus, the regularization term would be $B = \sum\limits_{i,j\in\Omega} (x_i^T\beta - x_j^T\beta)^2$, and by adding this term to naive EN, we would have:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \|y - \mathcal{X}\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 + \lambda_3\|D\beta\|_2^2 \tag{3.8}$$

where $B = \|D\beta\|_2^2$, and D shows the difference matrix with $\|\Omega\|$ rows, each of which is a vector obtained by difference of image pairs in neighboring frame set Ω. Adding the term $\|D\beta\|_2^2$ to Equation (3.7), intuitively penalizes sudden prediction change between neighboring frames. Equation (3.8) is further simplified as:

$$\hat{\beta^*} = \underset{\beta^*}{\operatorname{argmin}} \|\tilde{y} - \tilde{\mathcal{X}}\beta^*\|_2^2 + \gamma\|\beta^*\|_1 \tag{3.9}$$

where

$$\tilde{\mathcal{X}} = (1+\lambda_2)^{-1/2}\begin{pmatrix}\mathcal{X}\\ \sqrt{\lambda_2}I\\ \sqrt{\lambda_3}D\end{pmatrix} \quad \text{and} \quad \tilde{y} = \begin{pmatrix}y\\0\\0\end{pmatrix} \tag{3.10}$$

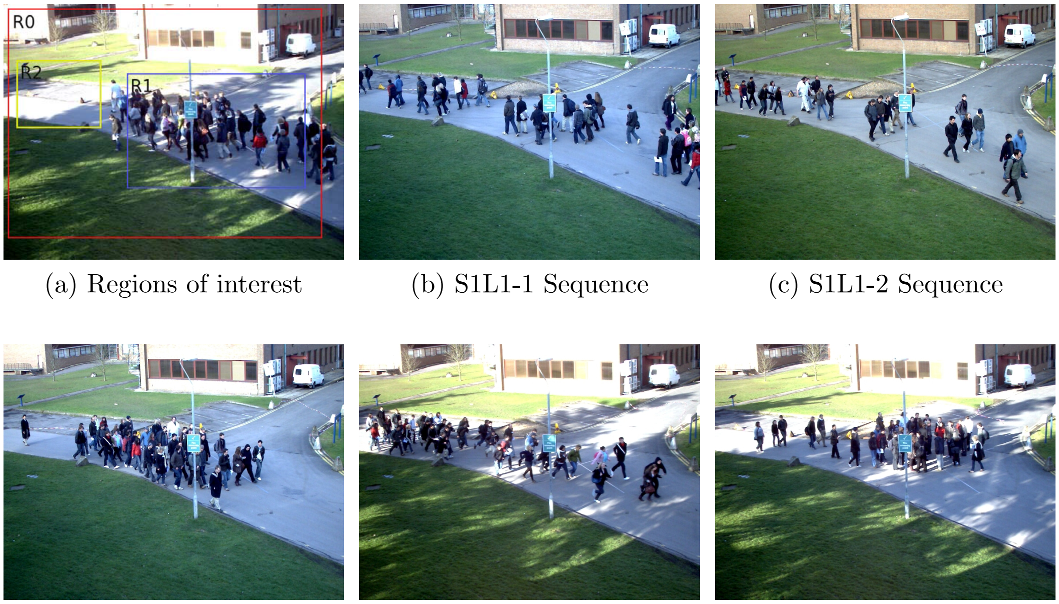
Here $\tilde{\mathcal{X}} \in R^{(n+m+\|\Omega\|)\times m}$ and $\tilde{y} \in R^{(n+m+\|\Omega\|)\times 1}$. We notice that (3.10) is a l_1-minimization problem, and can be effectively solved by Homotopy or DALM methods.

In the proposed framework the dataset is firstly partitioned into training and test sets. The N_{tr} training samples consists of l labeled samples as $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^{l}$ and u unlabeled data points as $\mathcal{U} = \{x_j\}_{j=l+1}^{l+u}$, where $N_{tr} = l + u$ $(l << u)$. The user manually annotates few l images, and remaining u unlabeled training data and N_{ts} unlabeled test data will be annotated automatically using SSEN. We iteratively select a sequential portion of unlabeled data and estimate their counts, and the most confident recently labeled instances with predicted labels are added to $\mathcal{L}$. SSEN is re-trained on the updated training set and all the steps are repeated till all the training samples being annotated.

It should be highlighted that how to select the data to be labeled, plays a key role in improving the annotation performance and saving human-labor. It is believed that given a fixed number of labeling budget, the most representative frames (in the sense of covering different crowd densities and/or counts) are the most useful ones to label. This brings in a chicken-and-egg problem [23], without labeling all frames, how does one know which ones are representative? Intuitively, the diversity between the selected frames should be as large as possible. Therefore, we employ a simple but effective way; we perform a k-means clustering on the samples and randomly select equal number of frames from each class as the labeled training data. The experiments validate the effectiveness of this selection paradigm. For instance Algorithm 3.1 shows the training procedure of labeling 5000 images for our algorithm.

Algorithm 3.1 Training of SRP-Count Algorithm

```
Initialize:
Dictionary $X$ with Gist; $N_{total} = 5,000, N_{labeled} = 0, i = 0, \mathcal{L} = \{\}, Failed = \{\}$
while $N_{labeled} <= N_{total}$ do
    Define $\mathbf{g} = [x_i \dots x_{i+1000}]$
    Partition $\mathbf{g}$ to train part $\mathbf{Tr}$ and test part $\mathbf{Ts}$
    Perform K-means clustering on $\mathbf{Tr}$ to find subset $\mathbf{l}$; Ask user to annotate $\mathbf{l}$ ; Find $\mathbf{u} = \mathbf{Tr} \setminus \mathbf{l}$
    Update
        $N_{labeled} = N_{labeled} + N_{\mathbf{l}}$
        $\mathcal{L} = \mathcal{L} \cup \mathbf{l}$
    Train SSEN on $\mathbf{l}$ and $\mathbf{u}$ using Equation (3.8) to find weights $\beta$
    Predict counts on $\mathbf{Ts}$ as $y = x^T \beta$
    Select most confident samples in $\mathbf{Ts}$ using no sudden change rule of p-frames; Name them as $\mathcal{S}$, Find $\mathcal{F} = \mathbf{Ts} \setminus \mathcal{S}$
    Update
        $\mathcal{L} = \mathcal{L} \cup \mathcal{S}$
        $N_{labeled} = N_{labeled} + N_{\mathcal{S}}$
        $Failed = Failed \cup \mathcal{F}$
        $i = i + 1000$
end while
Train Elastic Net on $\mathcal{L}$ using Equation (3.7) to find weights $\beta$
Predict counts on $Failed$ as $y = x^T \beta$
$\mathcal{L} = \mathcal{L} \cup Failed$
```



(a) Regions of interest (b) S1L1-1 Sequence (c) S1L1-2 Sequence

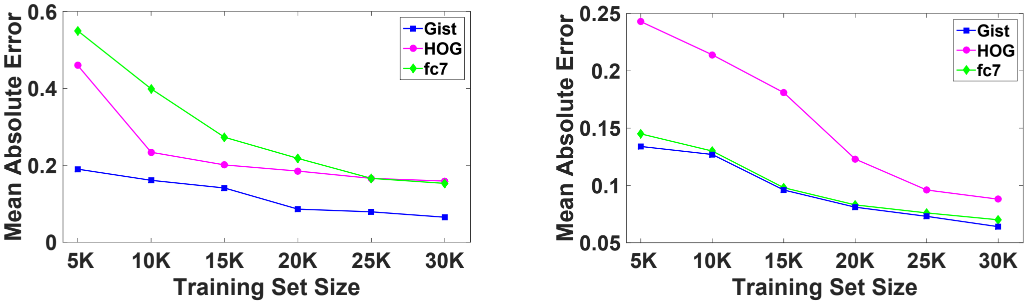
(d) S1L2-1 Sequence (e) S1L3-1 Sequence (f) S1L3-2 Sequence

Figure 3.7: Sample images from different sequences of the PETS 2009 pedestrian dataset

3.3.2 Experiments

Datasets: We conduct extensive experiments to validate our proposed method in different scenarios. In addition to the large-scale UCSD dataset [21], another crowd counting benchmark dataset is used for evaluations. The PETS 2009 dataset [1] has been captured under 7 fps with the image size 384 × 288. It contains 3 parts of multi-view sequences containing pedestrians walking in an outdoor environment, among which part S1 concerns people counting and density estimation. The S1 part has some sequences namely S1-L1, S1-L2 and S1-L3, and the task is to report the count within some provided ROIs. Some sample images of different sequences, along with the ROIs (R0, R1, and R2) are shown in Figure 3.7. To have a fair comparison, we follow the same settings in the literature for these sequences, which can be seen in Table 3.4. We notice that PETS 2009 is not a large dataset, and we exploit it for the sake of comparison, since a wide variety of methods have been evaluated on it.

Evaluation metrics: For the quantitative performance evaluation, we



(a) Peds1

(b) Peds2

Figure 3.8: MAE rate of SRP-Count method versus training size using different feature descriptors

use three classical error measures including MAE, MSE and MDE as defined in Equation (3.3). In the experiments of the UCSD dataset, the training set is constructed by randomly selecting images and this random selection process is repeated 10 times, and we report the average recognition rates for all the competing methods. For the PETS 2009 dataset, we follow the settings offered in Table 3.4, and there is no random selection process.

Supervised Learning: As explained, SRC needs a large enough dictionary of training samples to achieve great performance. To verify the effectiveness of involving more training data in this framework, we change the size of training set in the UCSD dataset. In this experiment we assume that all selected images are labeled, which has already been performed for the previously proposed RCS-Count method. We randomly select

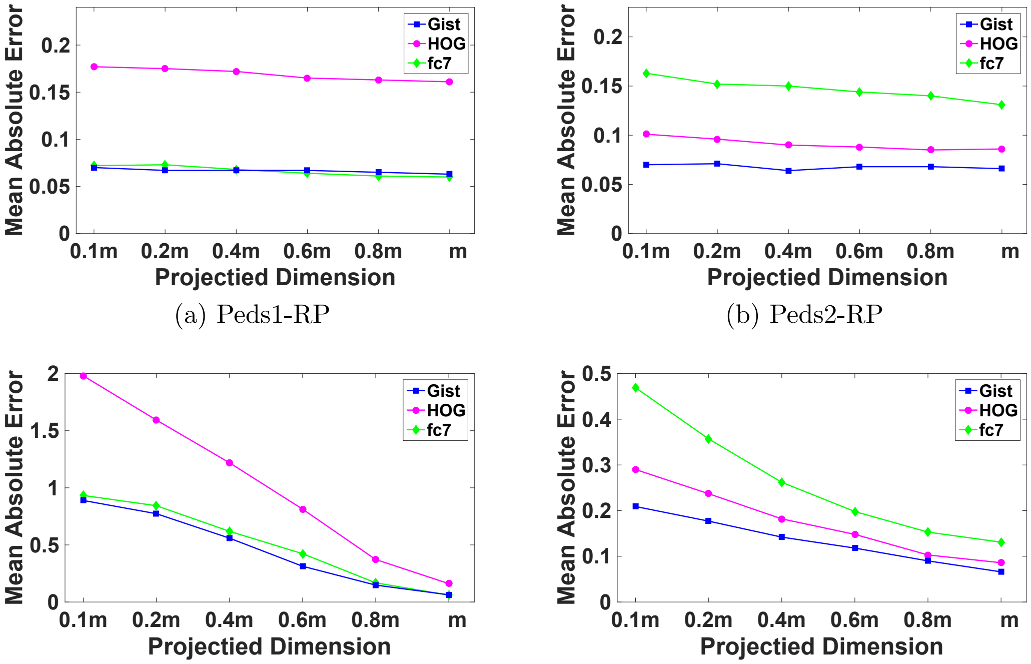
Table 3.4: The training and test frames of different sequences of the PETS 2009 dataset

Scenario	Test Set	ROIs	Training Set	N
S1L1-1	13-57	R0, R1, R2	13-59, 13-59F, 14-03, 14-03F	1308
S1L1-2	13-59	R0, R1, R2	13-57, 13-57F, 14-03, 14-03F	1268
S1L2-1	14-06	R1, R2	13-57, 13-57F, 13-59, 13-59F, 14-03, 14-03F	1750
S1L3-1	14-17	R1	13-57, 13-57F, 13-59, 13-59F, 14-03, 14-03F	1750

$\{5,000; 10,000; 15,000; 20,000; 25,000; 30,000\}$ training samples in each of the Peds1 and Peds2 datasets, and use the rest for the testing. Additionally, we compare the performance of different feature descriptors including Gist, HOG and deep features a.k.a fc7. Figures 3.8a and 3.8b show the MAE versus training set size on the Peds1 and Peds2 datasets, respectively. Using the same settings in Subsection 3.2.2, the Gist descriptor would be 512-dimensional for all evaluated datasets; however, in the case of HOG, we would have $2,160$ and $8,424$-dimensional feature vectors for the UCSD and PETS datasets, respectively. Besides, fc7 has 4096 dimensions for all the datasets.

As expected when we have larger training sets, the errors are smaller and all the descriptors perform well. In particular, Gist has the best overall performance over all training sizes in both datasets, and it is closely followed by HOG and fc7. Interestingly deep features trained on a totally different network obtained very competitive results, and it confirms the previous studies that these features can be reused in other tasks than classification. So, we can say in the proposed SRP-Count if the number of images is sufficiently large and the sparsity is properly harnessed, the choice of feature representation is no longer critical. These results also suggest that instead of re-training a counting model for an unseen scene, we could benefit from the deep features learnt from other tasks or preferably from some crowd scenes.

Next, we demonstrate the role of dimensionality reduction on the performance of the proposed SRP-Count method. In this experiment we randomly choose $30,000$ training images from the Peds1 and Peds2 datasets, and use the rest for testing. Using the projection matrix of PCA and RP, we gradually reduce the dimensionality of features from m to $0.1m$, and perform the classification with the low-dimensional features using Equation (3.6). Figure 3.9 illustrates the MAEs of different feature descriptors versus feature dimension, using PCA and RP for both datasets. It is observed that if we use RP, the error rates are less affected in lower dimensions, compared to PCA. This is mainly due to the fact that PCA is dependent on the omitted eigenvalues, while RP preserves the similarity of feature vectors well even in extremely low dimen-



(a) Peds1-RP

(b) Peds2-RP

(c) Peds1-PCA

(d) Peds2-PCA

Figure 3.9: MAE rate of SRP-Count method by different feature descriptors, versus feature dimension using RP and PCA

sions. The projected low-dimensional features by RP, retain their original performance even with 90% reduction in the original dimension. In particular, Gist still performs the best across all the reduced dimensions in both datasets. By adopting RP at the test time, the l_1-minimization problem (3.6) is solved very fast, without losing too much accuracy. Besides, the low computational complexity and simple nature of RP, makes it a great candidate to be adopted for a real-time crowd counting method.

We compare our method with regression-based counting methods in these large training sets. For our method, we use the Gist descriptor to represent images, and employ RP to reduce the dimension to just 10% of the original dimension. For regression-based methods, following the literature [21] three types of low-level features including segment, edge and texture are extracted from crowd segments, perspective normalised, and fed to regression models such as KRR [6], SVR [142] and GPR [21]. The MAE and MDE rates on different training sizes of the Peds1 and Peds2 datasets are summarized in Table 3.5.

Table 3.5: The error rates of different methods on the UCSD dataset

	Method	MAE				MDE			
		5K	15K	25K	30K	5K	15K	25K	30K
Peds1	**SRP-Count**	0.19	0.14	0.07	0.06	0.09	0.08	0.06	0.05
	KRR [6]	4.02	7.39	9.11	7.88	0.22	0.29	0.30	0.24
	SVR [142]	4.44	6.87	5.10	5.44	0.23	0.25	0.24	0.23
	GPR [21]	4.43	6.98	6.69	5.51	0.25	0.28	0.28	0.27
Peds2	**SRP-Count**	0.13	0.09	0.07	0.06	0.11	0.09	0.07	0.05
	KRR [6]	1.82	3.00	3.37	2.39	0.23	0.26	0.30	0.22
	SVR [142]	2.10	2.76	1.86	1.99	0.20	0.23	0.19	0.18
	GPR [21]	1.61	2.46	1.90	1.82	0.18	0.22	0.16	0.15

The proposed SRP-Count shows superior performance across all training sizes in both datasets, and as expected and previously seen in Subsection 3.2.2, the regression-based methods cannot be generalized well to large training datasets. This observation is arised from different reasons. First, when more data are involved in these models, the regression functions are mostly unable to adequately capture the non-linearity imposed in the feature space. Second, when these regression models are applied to unseen density, their performance could be quite different from what we may anticipate from training data. Third, as Loy *et al.* [93] showed, the performance of these methods is highly dependant on the selected features, and the optimal feature set could be totally different in various scenes according to the crowd structure and density.

SRP-Count on Small Datasets: In the previous part, it is assumed that we have big data available and the train set is densely sampled, which makes the classification task much easier since all testing variations have already been seen. But, to generalize the strength of our method we need to prove its robustness to small-sized datasets as well. According to previous results, we can claim that the proposed SRP-Count method achieves superior performances in large datasets; however, not all pedestrian datasets have such a large number of images. To evaluate the performance of our method on

Table 3.6: The error rates of different counting methods on the small UCSD dataset

Method	Peds1			Peds2		
	MAE	MSE	MDE	MAE	MSE	MDE
SRP-Count-Gist	0.95	5.13	0.10	0.80	2.53	0.09
SRP-Count-fc7	1.57	7.39	0.11	1.37	3.22	0.10
BPR [21]	3.65	7.41	0.12	1.77	2.54	0.12
KRR [6]	2.41	7.45	0.10	1.42	3.42	0.10
SVR [142]	2.46	7.75	0.10	1.80	3.50	0.10
GPR [21]	4.12	8.73	0.12	1.58	3.26	0.11
MORR [25]	2.49	8.68	0.10	1.33	2.72	0.11

smaller datasets, we utilize two datasets including the PETS 2009 dataset and the first 4000 frames of the UCSD dataset. Chan *et al.* [21] just annotated the first $4,000$ frames of the Peds1 and Peds2 sequences in the UCSD dataset, and most of crowd counting methods have been evaluated on this small sequence. Following [21], for the Peds1 the training set contains 1200 frames (frames $1401 - 2600$), and the remaining 2800 frames are used for testing. On the Peds2, the training set includes 1000 frames (frames $1501 - 2500$), with the remaining $3,000$ frames held out for testing. Table 3.6 lists the counting accuracy of our method versus the state-of-the-art regression-based methods. Although the error rates are not as small as the larger datasets, our proposed SRP-Count method still outperforms the regression-based method, even with deep features. Obviously, as the training set keeps getting bigger, SRC is coming to play a key role in providing better estimations and the superiority of SRC would be more remarkable.

For the PETS 2009 dataset, we follow the training/test settings explained in Table 3.4. We compare our method with the most successful methods of crowd counting participated in PETS 2009 competition, including regression-based methods such as [20], which all have been selected by Ferryman *et al.* [45] according to exhaustive performance evaluations. Table 3.7 compares the MAE of all competing methods on different video sequences over the corresponding

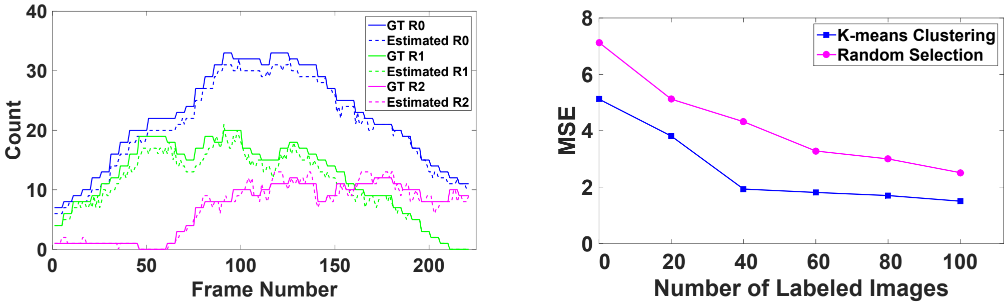
Table 3.7: The error rates of different counting methods on the PETS 2009 dataset

Method	Sequence								
	S1L1-1			S1L1-2			S1L2-1		S1L3-1
	R0	R1	R2	R0	R1	R2	R1	R2	R1
SRP-Count-Gist	1.30	1.23	0.71	1.01	0.70	0.99	1.95	1.90	0.86
SRP-Count-fc7	2.10	2.14	1.10	2.08	0.96	1.30	2.50	2.63	1.20
Chan *et al.* [20]	2.46	2.28	0.99	1.41	0.69	1.23	5.39	4.18	0.88
Alahi *et al.* [4]	–	–	–	4.20	2.30	1.87	6.50	4.00	0.90
Albiol *et al.* [5]	1.41	–	–	1.77	–	–	1.94	–	1.36
Choudri *et al.* [28]	1.29	2.23	0.70	3.26	3.18	1.04	3.70	4.17	0.67
Patzold *et al.* [109]	2.75	2.58	1.38	2.35	1.58	1.58	6.37	6.08	4.70
Conte *et al.* [30]	1.38	2.14	7.60	1.14	0.80	0.87	2.18	3.25	2.95

ROIs. We evaluate Gist and fc7 features, while fixing the projected dimension as 10% of the original dimension. The proposed SRP-Count method using Gist descriptor is superior or competitive to other methods, and especially shows promising performance on very dense crowd sequences such as S1L2-1 and S1L3-1. Even SRP-Count using deep features learnt on classification task, outperform hand-crafted features and/or carefully designed counting methods. Figure 3.10a shows the estimated count by SRP-Count method versus ground-truth on S1L1-1 sequence of the PETS 2009 dataset, over different ROIs. The result of this figure was generated according to Table 3.7, using 10% of the original dimension of Gist feature descriptor.

Semi-supervised Learning: In all of the above experiments on the UCSD dataset, we assumed that we have sufficient labeled training samples, which have been provided through manual annotation. As explained, we adopt SSEN to significantly reduce the amount of manual annotation, and make our method much more applicable in practice.

Suppose that we need to have atleast $5,000$ labeled images for the SRC, and we just have first $g = 2,000$ frames annotated. Data is split into training and test portions, each of which include N_{tr} and N_{ts} images as $g = N_{tr} +$



(a) (b)

Figure 3.10: The performance of SRP-Count method (a) Estimated count versus ground-truth on S1L1-1 sequence of the PETS 2009 dataset (b) Inductive MSE rate of SSEN on the Peds2-UCSD dataset using different labeled data selection methods

N_{ts}. We choose l labeled samples from the available N_{tr} training images by performing k-means clustering on the samples, and the rest of samples $g - l$ remain unlabeled. We evaluate SSEN by the transductive learning (test with unlabeled data in the training portion on $u = N_{tr} - l$ images), and inductive inference (test with the unlabeled data in the test portion on N_{ts} images); however, the latter is more important for us. Table 3.8 shows the MSE on both transductive and inductive cases on the Peds1 and Peds2 datasets, with $N_{tr} = 800, N_{ts} = 1200, l = 100$ while changing the number of unlabeled images u. It is evident from the results that using more unlabeled data greatly helps to reduce the errors. Also, as expected we have smaller transductive errors in both datasets.

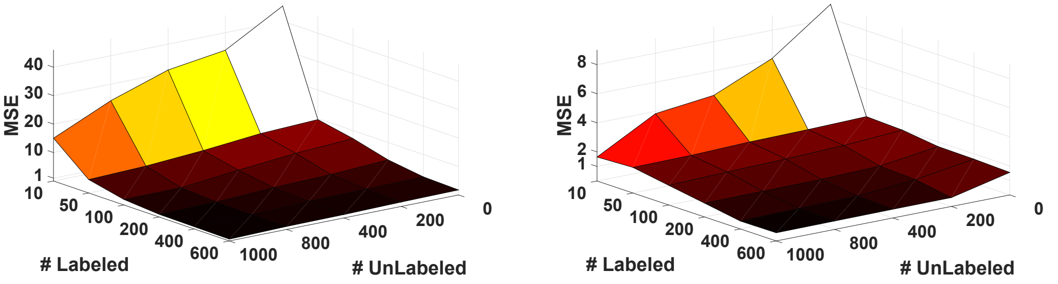
We expand this experiment and use the first $g = 4,000$ annotated images of the UCSD dataset, and following [21] we use $N_{tr} = 1200$ and $N_{ts} = 2800$ training and test images, respectively. We review the effect of increasing both labeled and unlabeled data by measuring the inductive MSE on N_{ts} test images. In this experiment, we use Gist descriptor to represent images, and RP to reduce the dimension to 10% of the original dimension. The optimal values of SSEN parameters are found by 5-fold cross validation. We choose different numbers of labeled images including $\{10, 50, 100, 200, 400, 600\}$, given unlabeled set $\{0, 200, 400, 800, 1000\}$, and illustrate the results on Figures 3.11a

Table 3.8: The MSE rate of SSEN on the UCSD dataset

	Peds1		Peds2	
	Transductive	Inductive	Transductive	Inductive
$u = 100$	1.03	6.51	0.84	1.88
$u = 300$	0.82	5.24	0.67	1.67
$u = 500$	0.55	3.43	0.50	1.50
$u = 700$	0.34	2.33	0.42	1.40

and 3.11b for the Peds1 and Peds2 datasets, respectively. We observe that when the number of labeled samples is small, increasing the number of unlabeled samples remarkably improves the prediction performance, which means the manual labelling work can be greatly reduced without losing the performance. For instance, given 50 labeled data the MSE is reduced by nearly 38% and 10%, when we increase the unlabeled data size from 200 to 800, in the Peds1 and Peds2 datasets, respectively. Besides, Figure 3.10b demonstrates how the selection process of the labeled images can affect the error rate. We perform an experiment on the Peds2 dataset, and increase the number of labeled samples from 0 to 100 while using $u = 400$ unlabeled images. We then compare the inductive MSE of random selection and k-means clustering techniques. Like before, we use Gist descriptor to represent images, and RP to reduce the dimension to 10% of the original dimension. The results suggest that clustering leads to better performance, while the random selection method cannot contribute to the semi-supervised learning that much since it blindly selects instances.

To provide a large training set for SRC, we start with a very small labeled training set and iteratively perform SSEN. Usually in each iteration, 60% and 40% of data are used for training and test parts, respectively. The training part is then partitioned into labeled and unlabeled parts under the ratio of 25% and 75%, respectively. In each iteration, the unlabeled samples in the test set are given predicted labels, and amongst them the most confident ones along with their predicted labels, are augmented to the initial labeled set. Ideally, these



(a) Peds1

(b) Peds2

Figure 3.11: The effect of increasing labeled and unlabeled data using SSEN on the UCSD dataset

selected samples can help to learn a better classifier for the next iteration. The learner is then re-trained and evaluated on the updated training and test sets, and the whole process iterates until all images are annotated. It should be emphasized that, if SSEN wrongly assigns labels to some unlabeled samples, the final inductive performance will be jeopardized due to the accumulation of mislabeled or badly-labeled data *i.e.*, the ones that their predicted and real counts are very different. So, to identify these samples, we measure the "confidence" of recently labeled samples, using the information of their neighbors. Specifically in pedestrian datasets, the unlabeled test set includes sequential frames, in which the pedestrian quantities of every p frames, are the same or change slightly. So, any significant change between the predicted counts of recently labelled neighboring frames can be considered as a mislabeled example. Once we identify these mislabeled data, we simply discard them while keeping the good ones intact. It is worth noting that, we do not try to re-label the identified mislabeled samples essentially in the next iteration. Probably most of the neighbors of a mislabeled image, have been correctly labeled and moved to the labeled part. So, the "sudden prediction change" trick cannot be exploited any more, due to to the lack of meaningful neighbors, in the next iteration. Alternatively, mislabeled samples are held out and annotated in the final iteration, which our model have been trained sufficiently with much more confident labels of their neighboring frames.

This iterative self-training labeling procedure is very beneficial, especially

Table 3.9: The error rates in supervised and semi-supervised SRP-Count method on the UCSD dataset

	Method	Feature	MAE				MDE			
			5K	15K	25K	30K	5K	15K	25K	30K
Peds1	Sup.	Gist	0.19	0.14	0.07	0.06	0.09	0.08	0.06	0.05
	SemiSup.	Gist	0.27	0.20	0.15	0.13	0.12	0.11	0.10	0.09
	Sup.	fc7	0.55	0.27	0.16	0.15	0.18	0.13	0.11	0.08
	SemiSup.	fc7	0.66	0.48	0.38	0.31	0.24	0.20	0.16	0.13
Peds2	Sup.	Gist	0.13	0.09	0.07	0.06	0.11	0.09	0.07	0.05
	SemiSup.	Gist	0.27	0.21	0.13	0.11	0.16	0.14	0.11	0.09
	Sup.	fc7	0.14	0.10	0.08	0.07	0.11	0.10	0.08	0.06
	SemiSup.	fc7	0.29	0.21	0.16	0.11	0.18	0.15	0.11	0.08

when we have limited labeling budget. However, one question maybe raised here; if iterative SSEN is a reliable method for predicting counts, why not use it to annotate (and meanwhile estimate the counts) the whole dataset, rather than utilizing SSEN just for preparing the rich training set of SRC? Although SSEN provides promising results on small test datasets in each iteration, it is more prone to erroneous prediction on large-scale datasets in comparison with SRC. Regression models, including SSEN, suffer from serious problems such as poor tractability and expensive training time, when they are generalized to large-scale datasets. The iterative SSEN also needs couple of training iteration which itself imposes more error and computational complexity. Equally important, SSEN relies on the assumptions that the temporal space is dense and abundant sequential unlabeled frames are available; however, this assumptions can be too stringent for many real-world scenarios when continuous video recording is not available.

Finally, we design an experiment to validate the accuracy of the self-labeled samples. We repeat our initial crowd counting experiments with large datasets, but instead of using the manually annotated training set, we use the training set provided by SSEN. Indeed, for the former we use full labeled data, while for the latter we just utilize $l = 600$ labeled images to start the iterative

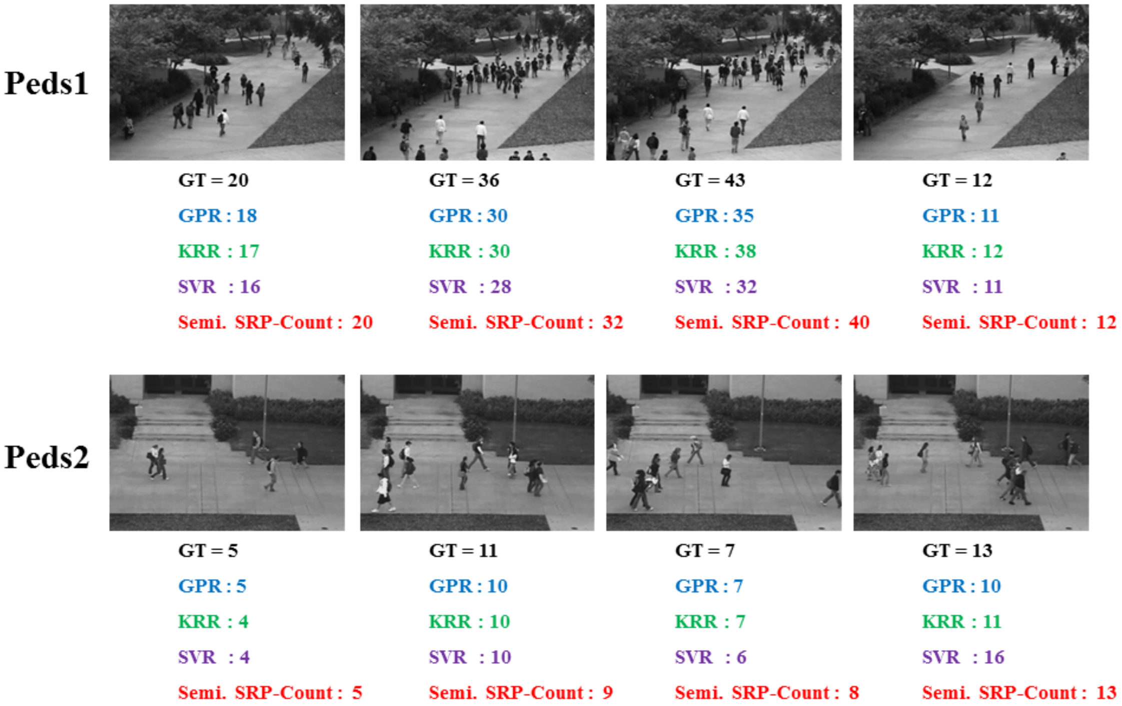


Figure 3.12: Comparison of counting performance between semi-supervised SRP-Count method and some regression-based methods on the UCSD dataset

SSEN, and gradually enlarge the training set. Table 3.9 summarizes the MAE and MDE of supervised and semi-supervised SRP-Count method across five different training set sizes on the Peds1 and Peds2 datasets. We adopt Gist and fc7 feature representations and use RP to reduce the dimension to 10% of the original dimension. To have a fair comparison, the training and test partitions of supervised and semi-supervised methods are kept similar in all the experiments. Not surprisingly, the counting accuracy is higher in the supervised SRC, and this can be explained by mislabeled examples and error accumulation; however, the performance has not been affected significantly, confirming the robustness of sparse representation to the noise introduced in the self-labeling process.

Figure 3.12 shows a comparison of the actual counting performance between semi-supervised SRP-Count method (using above mentioned settings) with some regression-based methods that are supervised and use all of the labeled training set. For this experiment, we selected 600 images initially by k-means clustering, to be labeled by the user and iterated SSEN is adopted provide $30K$ labeled images for the dictionary of SRC. We exploited Gist feature representations and RP to reduce the dimension to 10% of the original dimension. Although we exploit much less training data compared to other competing methods, the estimations are much more reliable due to powerful combination of SRC, RP and SSEN components.

3.3.3 Summary

In this section, we proposed an extremely accurate and scalable crowd counting method based on the integration of sparse representation-based classification and semi-supervised elastic net. Sparsity provides a powerful tool for inferring high-dimensional image data, that have complex low-dimensional structure and l_1-minimization offers computational tools to extract such structures, and helps to harness the semantic of data. The proposed SRP-Count method shows superior performance both in small and large pedestrian datasets; however, the advantage is much more remarkable in the latter. If the dictionary is rich and large enough to span the variations under test conditions, and sparsity is properly harnessed, the choice of image descriptors is no longer critical. We demonstrated that the learnt features from a CNN trained for object recognition task, can be reused in our framework for the purpose of crowd counting. RP also allows us to remarkably reduce the number of original features, without a significant loss in the accuracies. In addition, in order to provide the labeled training set with sufficient diversity, a semi-supervised elastic net model is employed to enable image annotation with just a few labeled images through exploiting the sequential information of readily available vast quantity of unlabeled data. We should mention that the proposed SRP-Count method achieved state-of-the-art crowd counting results at the publication date.

Chapter 4

Joint Feature Selection using Low-rank Dictionary Learning

4.1 Introduction

As explained in Subsection 2.6.2, feature selection using l_1-norm has been extensively discussed in [166], [39] and the results indicate good performance of these methods when spurious features exist along with relevant features. Thenceforth, different sparsity constraints have been introduced for feature selection. Most recently, Yan *et al.* [145] introduced the SRC measurement criterion into feature selection, and designed a joint sparse discriminative feature selection (JSDFS) method. Based on the assumption of SRC, JSDFS selects a subset of features which minimize intra-class reconstruction residual, and simultaneously maximize inter-class reconstruction residual in the subset of selected features. Although the so called JSDFS method could achieve promising results, it is well-known that SRC suffers from major drawbacks such as low discrimination and high computational complexity due to using all the training samples, and sensitivity to outliers, noisy observations (*e.g.*, occluded and/or corrupted), and illumination variations [150]. As a result, the reconstructive relationship of samples could not be well persevered by JSDFS method, and the selected features are not discriminant and robust enough.

To overcome the drawbacks associated with the SRC algorithm and to select the optimal subset of features, we propose a *Joint Feature Selection with Low-rank Dictionary Learning* (JFS-LDL) method, that is illustrated in Fig-

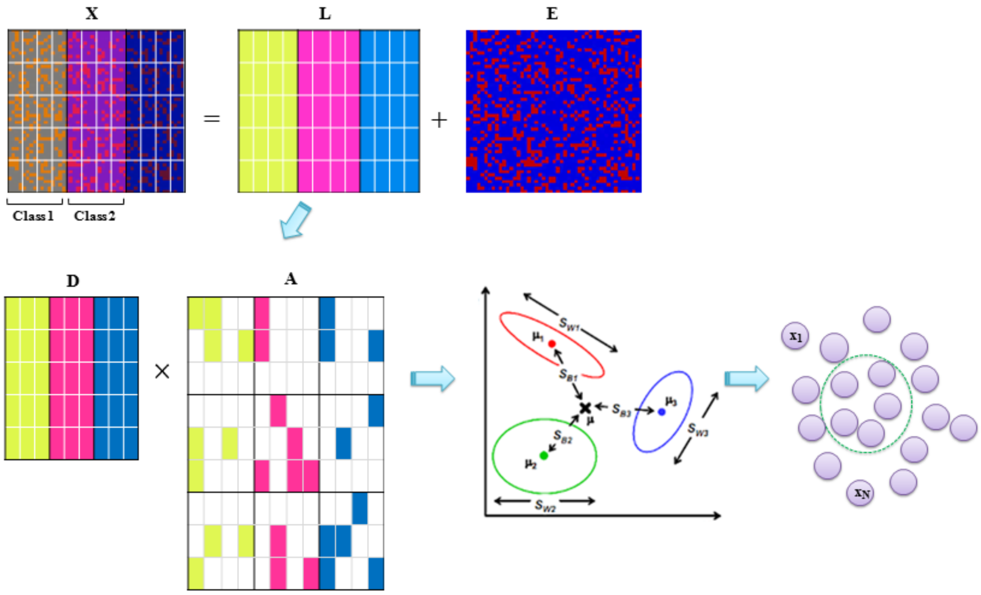


Figure 4.1: The overview of proposed JFS-LDL

ure 4.1. The proposed method selects features that simultaneously preserve the discriminative information and also the sparse reconstructive relationship of the data. To do so, we benefit from the integration of low-rank matrix recovery and Fisher discrimination dictionary learning to learn discriminative yet robust sparse representations form possibly noisy data. Then, the importance of a feature subset is evaluated by the ratio of intra-class to inter-class reconstruction residual in the selected subset. By incorporating $l_{2,1}$-norm minimization into the selection objective function, we are able to consider the correlation and interaction of features and select the most discriminative features from the whole feature space, all at once. We explain these components in more details in the following sections.

4.2 Low-rank Dictionary Learning using Fisher Discrimination

Yan *et al.* [145] showed that SRC measurement is a useful sparsity criterion to be used for selecting a subset of features that preserve the sparse reconstructive relationship of the data. However, SRC was shown to suffer from high computational cost and inadequate capability of discrimination. These

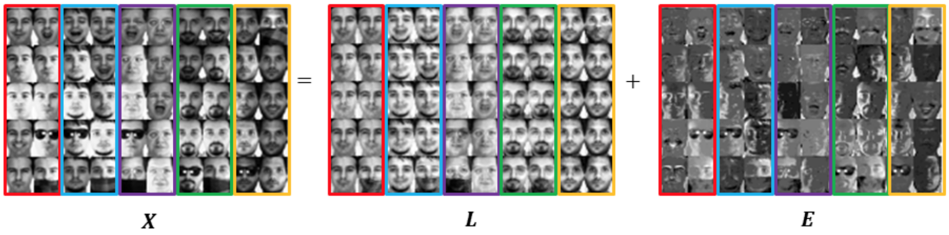


Figure 4.2: Low-rank decomposition on the AR dataset

issues can be effectively addressed by learning a smaller-sized discriminative dictionary from training images that can also increase the feature selection performance. However, in the real-world applications, images are not collected under well-controlled settings and could be easily contaminated by nuisance factors, such as occlusion, corruption, disguise, pose and lighting variations, and so on. In such cases, the performance of discriminative dictionary learning methods would be degraded significantly. To alleviate the effect of nuisance factors and to learn robust representation from contaminated observations, we incorporate low-rank matrix recovery into discriminative dictionary learning framework.

Denote by $X = [X_1, X_2, \ldots, X_K] \in R^{m \times N}$ a set of training samples, where X_i is the subset of the training samples from the i^{th} class, m is the feature dimension, N is the total number of training samples, and K is the number of classes. To efficiently remove sparse noises such as occlusion, illumination changes, pixel corruption from the observations, we use low-rank matrix recovery [16], and decompose the data matrix X_i as follows:

$$\min_{L_i, E_i} \|L_i\|_* + \lambda \|E_i\|_1 \quad s.t. \quad X_i = L_i + E_i \quad \forall i = 1 \ldots K \tag{4.1}$$

where $\|.\|_*$ denotes nuclear norm. As dicussed in Section 2.5, inexact ALM method [86] can be used to efficiently solve Equation (4.1). Figure 4.2 illustrates the result of low-rank decmposition on the AR face dataset [99], which is known for different facial expressions, illumination conditions and disguises including scarf and sunglasses.

The samples in class i are linearly correlated in many situations and low-rank matrix recovery reveals the structural information of each class and makes the training samples of that class more correlated; hence, the intra-class diversity is reduced. It is widely believed that L_i has better representation ability than X_i, since the sparse noise has been removed. Nevertheless, in some classification tasks such as face recognition, the images from different classes typically share common and correlated features (*e.g.*, for face images, the locations of eyes and nose are shared in different subjects). Since the derived matrix L_i might not contain sufficient discriminating information, we use a discriminative dictionary learning method to provide discriminating ability to the derived low-rank representations.

We aim to learn dictionary D and sparse coding coefficents A from low-rank representation of images. Denote by $D = [D_1, D_2, \ldots, D_K]$ the structured dictionary, where D_i is the class-specified sub-dictionary associated with the i^{th} class. With all the L_is found by (4.1), we get the whole low-ranked training samples as $L = [L_1, L_2, \ldots, L_K]$. Then, we adopt Fisher discrimination dictionary learning (FDDL) [150] objective function due to its strong discrimination power, and use L as the new representation of training samples. Denote by A, the sparse coding coefficient matrix of L over D, the structured dictionary D should have the capability to represent the sparse coefficients, $i.e., L \approx DA$. We can write A as $A = [A_1, A_2, \ldots, A_K]$, where A_i is the representation matrix of L_i over D. The dictionary D should have powerful reconstruction and discrimination capabilities to represent low-rank representation of images. So, the objective function of low-rank Fisher discrimination dictionary learning (LR-FDDL) model would be as:

$$J(D, A) = \underset{D,A}{\operatorname{argmin}} \left\{ r(L, D, A) + \lambda_1 \|A\|_1 + \lambda_2 f(A) \right\} \tag{4.2}$$

where $r(L, D, A)$ is the discriminative fidelity term, $\|A\|_1$ is the sparsity constraint, $f(A)$ is the discrimination constraint imposed on the coefficient matrix A, and λ_1, λ_2 are scalar parameters.

To learn a representative and discriminative dictionary, the structured dictionary D should be able to well represent low-rank representation of samples from any class. Also, L_i should be well represented by the associated sub-dictionary D_i, but not so well by other sub-dictionaries $D_j\,, j \neq i$. Following

FDDL notations, A_i can be written as $A_i = [A_i^1\,; \ldots\,; A_i^j\,; \ldots\,; A_i^K]$, where A_i^j is the representation coefficients of L_i over sub-dictionary D_j. Then, the discriminative fidelity term is defined as:

$$r(L_i, D, A_i) = \|L_i - D\,A_i\|_F^2 + \|L_i - D_i\,A_i^i\|_F^2 + \sum_{\substack{j=1\\j\neq i}}^{K} \|D_j\,A_i^j\|_F^2 \tag{4.3}$$

If we only require D to represent L_i well, *i.e.*, with only the first penalty $\|L_i - D\,A_i\|_F^2$, then $D_i A_i^i$ may deviate much from L_i so that D_i could not well represent L_i. This problem can be solved by adding the second penalty $\|L_i - D_i\,A_i^i\|_F^2$. Nonetheless, other sub-dictionaries may also be able to well represent L_i, reducing the discrimination capability of D. With the third penalty $\|D_j\,A_i^j\|_F^2$, the representation of D_j to L_i, $j \neq i$, will be small, and the proposed discriminative fidelity term could meet all our expectations.

To have more discrimination in the model, we make the coding coefficient of L over D, *i.e.*, A, be discriminative. This is achieved by defining discrimination constraint $f(A)$ as follows:

$$f(A) = tr\big(S_W(A)\big) - tr\big(S_B(A)\big) + \eta\|A\|_F^2 \tag{4.4}$$

Here, $S_W(A)$ and $S_B(A)$ are intra-class and inter-class scatter matrices of sparse coefficients A, which are defined as:

$$S_W(A) = \sum_{i=1}^{K} \sum_{a_k \in A_i} (a_k - m_i)(a_k - m_i)^T \tag{4.5}$$

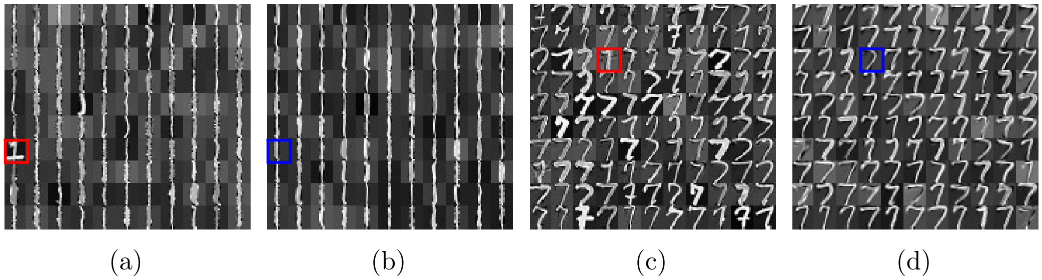
$$S_B(A) = \sum_{i=1}^{K} n_i (m_i - m)(m_i - m)^T$$

where m_i and m are the mean vectors of A_i and A respectively, and n_i is the number of training samples in class i.

By incorporating Equations (4.3) and (4.4) into Equation (4.2), we have the following low-rank Fisher discrimination dictionary learning model:

$$J(D, A) = \underset{D,\,A}{\operatorname{argmin}} \sum_{i=1}^{K} \Big(\|L_i - D\,A_i\|_F^2 + \|L_i - D_i\,A_i^i\|_F^2 + \sum_{\substack{j=1\\j\neq i}}^{K} \|D_j\,A_i^j\|_F^2 \Big) + \tag{4.6}$$

$$\lambda_1 \|A\|_1 + \lambda_2 \Big(tr(S_W(A) - S_B(A)) + \eta\,\|A\|_F^2 \Big)$$



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

Figure 4.3: The learnt sub-dictionaries of digits 1 and 7 of the USPS dataset by (a),(c) FDDL model (b),(d) LR-FDDL model

Low-rank matrix recovery reduces the intra-class diversity in each class and this might affect the representation power of dictionary. However, incorporating Fisher discrimination on both class-specific representations and sparse coefficients provide enough discriminating ability into our framework. As a result, we can learn discriminative and robust representations from contaminated observations. We notice that the objective function (4.6) is not jointly convex in (D, A), but it is convex with respect to each of D and A, when the other is fixed; so, an alternating optimization algorithm is employed. The objective function (4.6) is similar to that of FDDL [150], except that we used L_i as the new representation of training images. So, we adopt the same optimization strategy and briefly review it in the following subsection.

To demonstrate the effect of low-rank matrix recovery, we compare the sub-dictionaries learned by the FDDL model [150] (X_i as input of model) and LR-FDDL (objective function (4.6), L_i as input of model) on the USPS handwritten digits dataset [66]. Figure 4.3 shows the sub-dictionaries for two digits 1 and 7. In LR-FDDL model, the variations in the shape, thickness and orientation have been significantly removed by sparse noise. Also, by reducing the intra-class diversity, dissimilarity between different classes would be increased, which means sub-dictionaries are more discriminant toward each other. For instance, digits 1 and 7 shown in red squares look very similar in the learnt sub-dictionaries of FDDL model; so, their sparse representations also resemble and they could be easily misclassified. The same pair is illustrated in the blue squares using LR-FDDL model, and we observe that they have better representative abilities and seem much more distinguishable.

4.2.1 Optimization of LR-FDDL

The objective function (4.6) can be divided into two sub-problems: updating A by fixing D, and updating D by fixing A. The procedures are iteratively implemented for dictionary D and sparse coefficients A.

First, assuming that D is fixed, the objective function (4.6) is further reduced to a sparse coding problem to compute $A = [A_1, A_2, \dots, A_K]$. We optimize A_i class-by-class and meanwhile, make all other $A_j\,(j \neq i)$ fixed. Thus, Equation (4.6) is simplified as:

$$J(A_i) = \underset{A_i}{\operatorname{argmin}} \left\{ r(L_i, D, A_i) + \lambda_1 \left\| A_i \right\|_1 + \lambda_2\, f_i(A_i) \right\} \tag{4.7}$$

where

$$f_i(A_i) = \| A_i - M_i \|_F^2 - \sum_{j=1}^{K} \| M_j - M \|_F^2 + \eta \left\| A_i \right\|_F^2 \tag{4.8}$$

where M_j and M are the mean matrices of class j and all classes, respectively, which are built by taking n_j mean vectors of m_j or m as their column vectors. It is shown in [150] that (4.7) can be rewritten as:

$$J(A_i) = \underset{A_i}{\operatorname{argmin}} \left\{ Q(A_i) + 2\tau \left\| A_i \right\|_1 \right\} \tag{4.9}$$

where $Q(A_i) = r(L_i, D, A_i) + \lambda_2\, f_i(A_i)$ and $\tau = \lambda_1/2$. Since $Q(A_i)$ is strictly convex and differentiable to A_i, the iterative projection method (IPM) [115] or improved approaches like FISTA [10] can be employed to solve Equation (4.7).

The next step is updating dictionary D, when A is held fixed. We update D_i class-by-class, by fixing all other $D_j\,(j \neq i)$. So, the objective function (4.6) is reduced to:

$$J(D_i) = \underset{D_i}{\operatorname{argmin}} \left\{ \| \hat{L} - D_i\, A^i \|_F^2 + \| L_i - D_i\, A_i^i \|_F^2 + \sum_{\substack{j=1 \\ j \neq i}}^{K} \| D_i\, A_j^i \|_F^2 \right\} \tag{4.10}$$
$$s.t. \quad \left\| d_c \right\|_2 = 1, c = 1, \dots, p_i$$

where d_c indicates each column of D_i and $\hat{L} = L - \sum_{j=1, j \neq i}^{K} D_j A^j$, that A^j is the coding coefficients of L over D_i and p_i is the number of atoms of the sub-dictionary D_i. Equation (4.10) can be rewritten as:

$$J(D_i) = \underset{D_i}{\operatorname{argmin}} \, \|\Gamma_i - D_i \, Z_i\|_F^2 \quad s.t. \quad \|d_c\|_2 = 1, c = 1, \ldots, p_i \tag{4.11}$$

where $\Gamma_i = [\hat{L} \, L_i \, 0 \, \ldots \, 0]$ and $Z_i = [A^i \, A_i^i \, A_1^i \, \ldots \, A_{i-1}^i \, A_{i+1}^i \, \ldots \, A_K^i]$, and 0 is a zero matrix with appropriate size based on the context. Equation (4.11) is a quadratic programming problem, which is solved using the algorithm propsoed in [152], where updates D_i atom by atom.

LR-FDDL converges since the two alternating optimizations are both convex. Yang *et al.* [148] also provided a simplified version of FDDL, in which $\|D_j \, A_i^j\|_F^2 = 0$ for $j \neq i$. The optimization algorithm of simplified FDDL can be considered as a spacial case of the general FDDL optimization framework. We refer the reader to [150] for more details on the optimization details and the convergence properties.

4.3 Joint Feature Selection

Our proposed JFS-LDL is as an embedded feature selection model, in which the procedure of feature selection is embedded directly in the training process. To find the optimal subset of features, we use the dictionary learning decision rule as the selection criterion. According to the characteristics of class-specific dictionary learning methods, a good feature subset is the one whose components could be well approximated by a linear combination of the other components in the same class, but not well by that of other classes. This can be achieved by minimizing the intra-class reconstruction residual and simultaneously maximizing the inter-class reconstruction residual in the subset of selected features.

So, we first define the intra-class (S_W^L) and inter-class (S_B^L) sparse scatter matrices as:

$$S_W^L = \frac{1}{N} \sum_{i=1}^{K} \sum_{j=1}^{n_i} \left[L_{i,j} - D_i \, A_{i,j}^i\right] \left[L_{i,j} - D_i \, A_{i,j}^i\right]^T \tag{4.12}$$

$$S_B^L = \frac{1}{N(K-1)} \sum_{i=1}^{K} \sum_{j=1}^{n_i} \sum_{\substack{s=1 \\ s \neq i}}^{K} \left[L_{i,j} - D_s \, A_{i,j}^s\right] \left[L_{i,j} - D_s \, A_{i,j}^s\right]^T$$

where n_i is the number of training samples in the i^{th} class, $L_{i,j}$ is the low-rank representation of $X_{i,j}$ *i.e.,* the j^{th} image from the i^{th} class. Denote by A_i^i the sparse coding coefficient of L_i over D_i ; $A_{i,j}^i$ implies its j^{th} column. Similarly, $A_{i,j}^s$ is the sparse coding coefficient of $L_{i,j}$ over D_s .

We aim to learn the projection (feature selection) matrix $P \in R^{m \times m}$ from given training data $X = [x_1, \ldots, x_N] \in R^{m \times N}$. During the learning process, in addition to optimizing the ratio of scatter matrices, we impose $\mathrm{l}_{2,1}$-norm on the projection matrix to encourage row-sparsity. It is worth noting that $\mathrm{l}_{2,1}$-norm has already been successfully applied in group Lasso [156], multi-task feature learning [90] and feature selection [104]. Because of $\mathrm{l}_{2,1}$-norm, most of the rows in the learnt projection matrix, shrink to zero and only d non-zero rows would remain. These non-zero rows indicate the d optimal features to be selected. This framework enables us to consider the correlation and interaction amongst features, and choose the optimal subset of features altogether. The low-dimensional representation of image x is obtained as $x' = P^T x$ where $x'(k) = x(k)$ if the k^{th} feature is selected, otherwise $x'(k) = 0$.

Hence, to optimally preserve the sparse reconstructive relationship of data and simultaneously achieving row-sparsity, the projection matrix P is found by the following optimization problem:

$$\min_P \frac{tr(P^T S_W^L P)}{tr(P^T S_B^L P)} + \beta \left\| P \right\|_{2,1} \tag{4.13}$$

Figure 4.4 shows an example of obtained P on the USPS handwritten digits dataset [66], in which many rows shrink to zero. The corresponding features to these zero rows are not important, and can be removed from the feature space. In this way, we can determine the optimal number of features automatically; however, if we are interested to determine the number of selected features manually, we can keep the rows with highest average values. The ratio trace [52] problem in Equation (4.13) is equivalent to the following problem:

$$\min_P tr(P^T S_W^L P) \quad s.t. \quad P^T S_B^L P = I \tag{4.14}$$

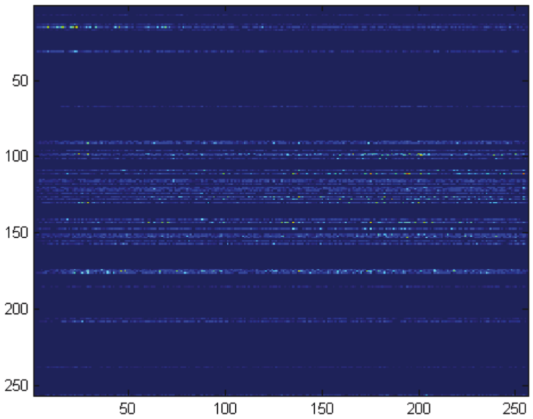


Figure 4.4: Sample projection matrix found by JFS-LDL on the USPS dataset

So, the objective function (4.13) is reformulated as:

$$\min_{P} tr(P^T S_W^L P) + \beta \left\| P \right\|_{2,1} \quad s.t. \quad P^T S_B^L P = I \tag{4.15}$$

Although the objective function (4.15) is convex, the constraint is not, and the problem needs to be reformulated to a more convenient form to be solved. Based on the Theorem 1 of [145], P can be obtained through the following two steps:

1. Solve the eigen-problem $S_W^L\, Y = \Lambda\, S_B^L\, Y$ to find Y

2. Find P which satisfies $D^T\, P = Y$

where Y is the matrix of generalized eigenvectors corresponding to $min(N, m)$ largest eigenvalues, Λ is a diagonal matrix whose diagonal elements are eigenvalues, and D is the structured dictionary. Finding a solution for P under $\mathrm{l}_{2,1}$-norm constraint such that $D^T P = Y$, is usually impossible. Therefore, a residue matrix E is introduced and the following problem is solved instead:

$$\min_{P,E} \left\| E \right\|_F^2 + \beta \left\| P \right\|_{2,1} \quad s.t. \quad D^T P + E = Y \tag{4.16}$$

We utilize an iterative inexact ALM to solve (4.16). The augmented Lagrangian function of (4.16) is defined as follows:

$$\mathcal{L}(P, E, M, \mu) = \left\| E \right\|_F^2 + \beta \left\| P \right\|_{2,1} + \frac{\mu}{2} \| D^T P + E - Y \|_F^2 + \left\langle M, D^T P + E - Y \right\rangle \tag{4.17}$$

where M is the Lagrange multipliers, and μ is a positive scalar. Inexact ALM alternatively updates the variables P and E by iteratively minimizing the augmented Lagrangian function $\mathcal{L}$. Algorithm 4.1 outlines the proposed JFS-LDL method, and the details of solving (4.17). In this algorithm, S is a diagonal matrix $S_{ii} = 1/2\|P_i\|_2$, where P_i is the i^{th} row of matrix P. It is worth pointing out that the convergence of inexact ALM, with at most two blocks has been well studied, and a proof to demonstrate its convergence property can be found in [86]. The learnt projection matrix by Algorithm 4.1, well preserves intra-class compactness and inter-class separability in the low-dimensional space. In addition, the proposed framework allows us to learn a discriminative and robust subspace, in which data can be easily separated.

Algorithm 4.1 JFS-LDL Algorithm

```
Input: Data matrix $X$
Output: Projection matrix $P$
Find low-rank representation $L_i$ of training samples $X_i$ for all $K$ classes by Equation (4.1).
Find dictionary $D$ and sparse coding coefficients $A$ by LR-FDDL by Equation (4.6).
Construct intra-class and inter-class sparse scatter matrices $S_W^L$ and $S_B^L$ by Equation (4.12).
Solve the eigen-problem $S_W^L\, Y = \Lambda\, S_B^L\, Y$ to find $Y$.
Initialize:$\beta = 10^{-6} \times \|\bar{D}\|_F^2$, $M = 0$, $\mu = 10^{-6}$, $max_\mu = 1.01$, $\rho = 2$.
Initialize:$P$ by solving linear equations $D^T P = Y$, and $E = 0$.
while $\left\|D^T P + E - Y\right\|_\infty < \epsilon$ do
    Update $E$ as: $E = \frac{1}{2+\mu}(-M + \mu Y - \mu D^T P)$
    Define $S$ as a diagonal matrix where $S_{ii} = \frac{1}{2\|P_i\|_2}$
    Update $P$ as: $P = (2\beta S + \mu D D^T)^{-1}(\mu D Y - D M - \mu D E)$
    Update $M$ as: $M = M + \mu(D^T P + E - Y)$
    Update $\mu$ as: $\mu = min(\rho\mu, max_\mu)$
end while
```

4.4 Time Complexity

The time complexity of Algorithm 4.1 is discussed as follows:

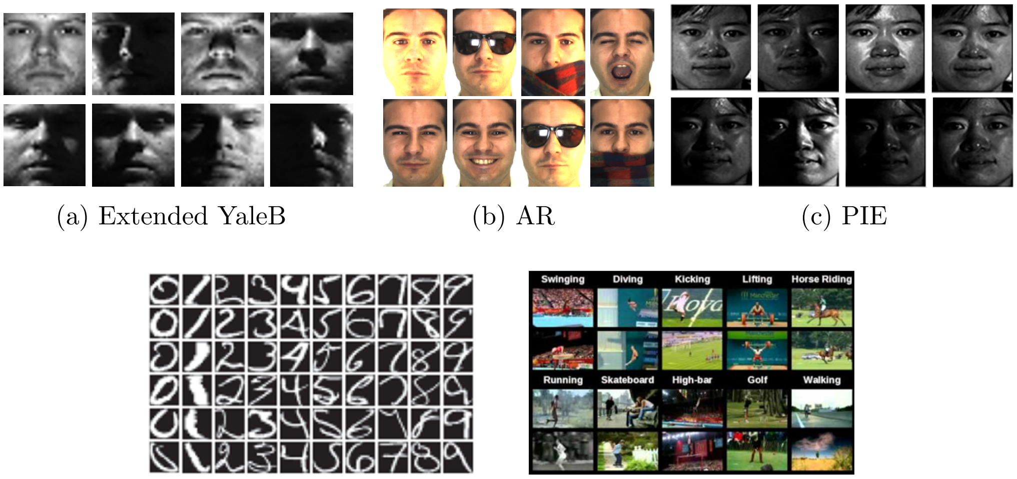
- To find low-rank representation of all the training samples of all K classes, we may use RPCA [16]; however RPCA is computationally expensive with complexity $O\big(min(m^2N, mN^2)\big)$ due to multiple iterations of SVD. Recently, several methods have been proposed for fast low-rank recovery and here, we utilize the accelerated version of robust orthonormal subspace learning (ROSL+) [123], which its complexity is bounded by $O\big(r^2(m+N)\big)$, where r is the rank of matrix L_i. Hence, the complexity of this step would be $O\big(K\bar{r}^2(m+N)\big)$, where $\bar{r}$ is the average value of rs for all K classes.

- The complexity of LR-FDDL to find dictionary D and sparse coding coefficients A, is similar to that of FDDL, consisting of updating sparse coefficients and sub-dictionaries. So, the overall time complexity this step is approximately $t\big(NO(m^2p^\epsilon)\ + \sum\limits_{i=1}^{K} p_iO(2mN)\big)$, where t is the total number of iterations of LR-FDDL, p_i is the number of i^{th} sub-dictionary atoms and $\epsilon \geq 1.2$ is a constant [151]. If we exploit the simplified version, it would have much lower time complexity than the original one. In that case, the overall time complexity is $t\big(\sum\limits_{i=1}^{K} n_iO(m^2p_i^\epsilon)\ + \sum\limits_{i=1}^{K} p_iO(mn_i)\big)$.

- The computational cost of constructing intra-class scatter matrix S_W^L and inter-class scatter matrix S_B^L is $\sum\limits_{i=1}^{K} n_iO(m^2+mp_i)$ and $\sum\limits_{i=1}^{K} n_iO\big((K-1)(m^2+mp_s)\big)$, where $s \neq i$, respectively.

- For eigen-decomposition step of finding Y, we exploit Lanczos algorithm [107] to compute the top $c = min(N, m)$ eigenvectors with complexity $O(cm^2)$.

- To find P, we utilize inexact ALM as shown in steps $6-10$ of Algorithm 4.1. The complexity is bounded by $O\big(\gamma(m^3+6m\mathcal{P}^2)\big)$, where $\mathcal{P} = \sum_i p_i$ and γ is the number of inexact ALM iterations.



(a) Extended YaleB

(b) AR

(c) PIE

(d) USPS

(e) UCF

Figure 4.5: Some example images of different datasets

4.5 Experimental Results

We conduct extensive experiments on several benchmark datasets to verify the effectiveness of the proposed JFS-LDL method in comparison with other feature selection methods, and validate its capability for different classification tasks including face recognition, handwritten digit recognition, and sport action recognition. For all the upcoming experiments we use nearest neighbor (NN) classifier unless another setting is mentioned specifically. Furthermore, we use the proposed JFS-LDL to perform cell counting in microscopic images, especially for breast cancer disease.

4.5.1 Parameter Selection

There are four parameters in the proposed JFS-LDL method that should be tuned. In low-rank decomposition step we have λ, for which the default parameter setting of inexact ALM is adopted. There are two parameters λ_1 and λ_2 in the LR-FDDL model, that we search them from a small set $\{0.001, 0.005, 0.01, 0.05, 0.1\}$. Changing the value of the regularization parameter β in projection learning step, would not change the accuracy and we set it as a small value such as 0.2. Based on our extensive experiment experience, we realized that the selection of these parameters is relatively independent of each other, and we use 5-fold cross validation to find the optimal values.

4.5.2 Object Classification

In object classification experiments, the training set is constructed by randomly selecting images, and this random selection process is repeated 10 times, and we report the average recognition rates for all the competing methods. The default number of dictionary atoms on each class is set as the number of training samples. The maximum iteration of all the iterative methods is set to 15.

- **Face Recognition**: We evaluate the performance of the proposed algorithm on three face recognition benchmark databases. In all the face recognition experiments, the raw pixels are used as the input of our method.

(a) Extended YaleB: The Extended YaleB [54] contains $2,414$ frontal face images of 38 human subjects under different illumination conditions. Each individual has around $59 \sim 64$ images, and we randomly select 32 images as training and use the rest for testing. All the face images are resized to 32×32.

(b) AR: The AR face dataset [99] consists of over $4,000$ frontal images from 126 individuals. For each individual, 26 face images are collected under different illumination, expression, and facial occlusion (disguise) in two separate sessions. As a standard evaluation procedure, we select a subset of $2,600$ images from 50 male and 50 female subjects in the experiments. Each face image is resized to 27×20. Focusing on the illumination and expression condition, we choose 7 unobscured (neutral) images from session 1 for training, and 7 images from session 2 for testing. Also, to verify the robustness of the proposed method to occlusion, we follow the "Sunglasses+Scarf" scenario [24], in which we consider the case where images with sunglasses and scarf are presented during training. We choose all 7 neutral images from the first session, and 2 corrupted images (one with sunglasses and the other with scarf) for training. We then use 17 images including 7 neutral images at session 2, plus the remaining 10 occluded images for testing.

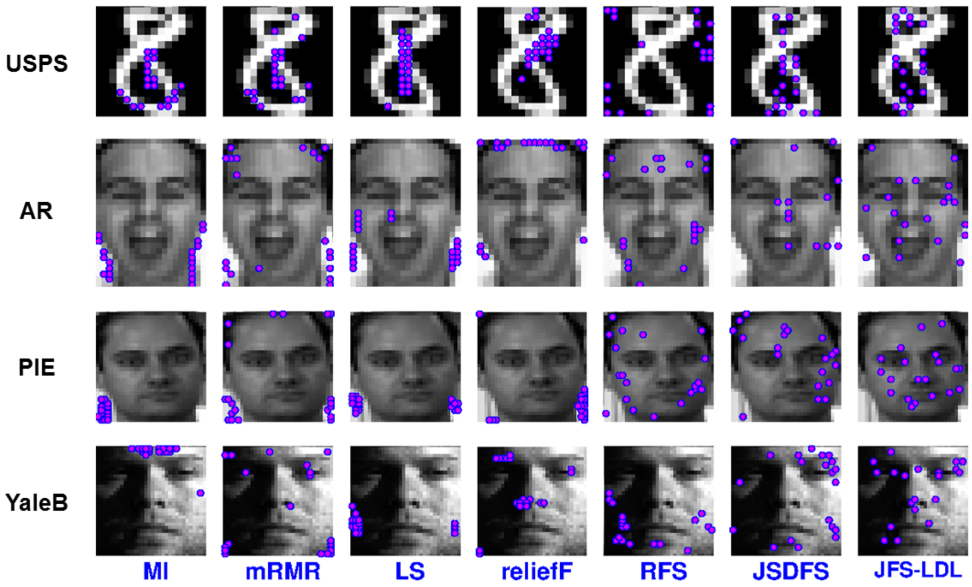


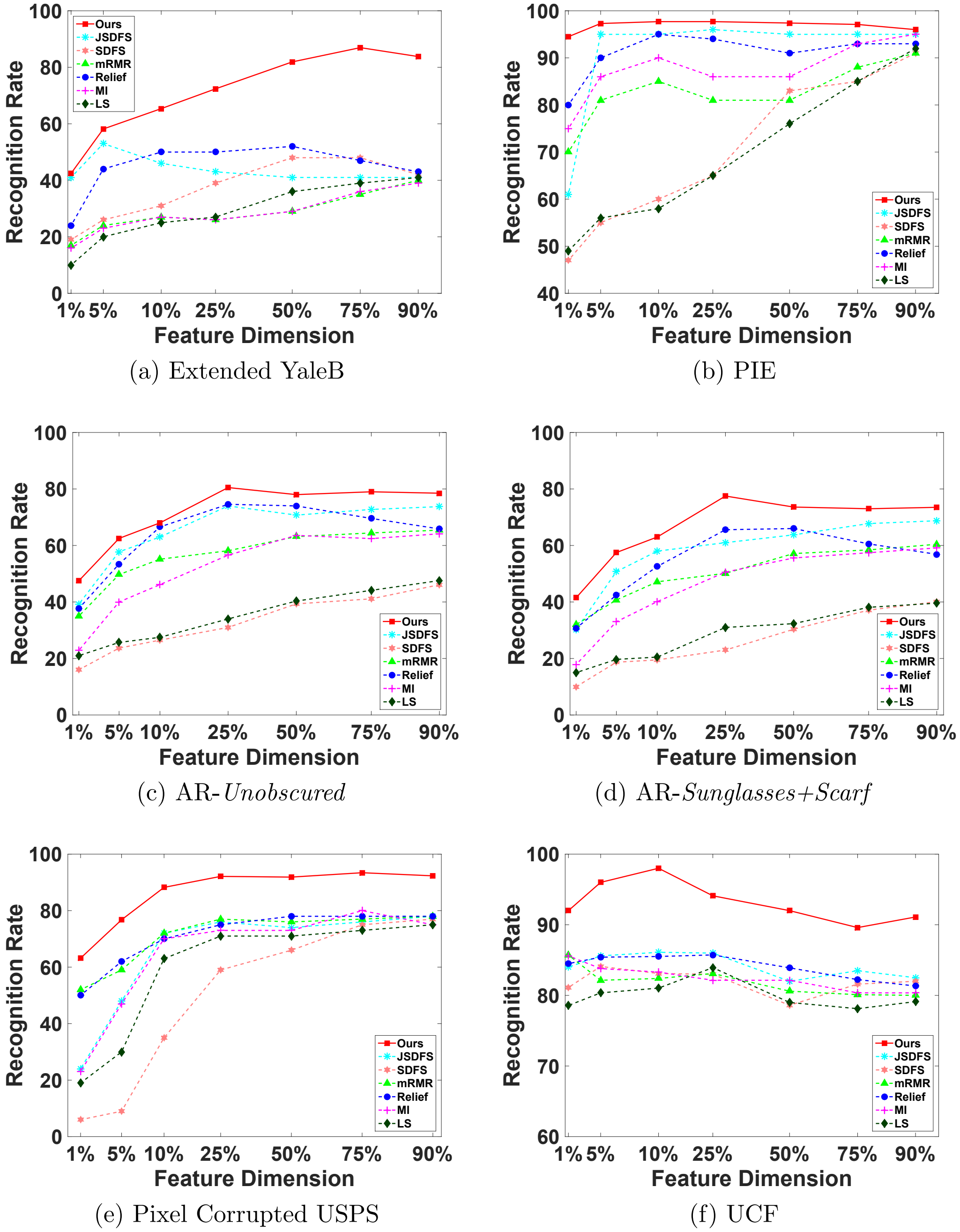
Figure 4.6: 20 Selected features (pixels) by different feature selection methods on various datasets

(c) CMU PIE: The CMU PIE [124] dataset consists of 41, 368 images of 68 subjects. For each person images are taken under different poses, expressions and illumination conditions. In our experiments, we just use the near frontal pose (C27) which leaves us about 100 face images for each individual, 30 of which are randomly chosen for training and the rest is used for testing. All Images are resized to 32 × 32.

- **Digit Recognition**: We perform handwritten digit recognition on the widely used USPS dataset [66], which has 7, 291 training and 2, 007 test images, each of size 16 × 16. Here, we use raw pixels to represent images, and set the number of sub-dictionary atoms to 200.
- **Action Recognition**: We also conduct action recognition on the UCF sport action dataset [114]. There are 140 videos in the UCF dataset, that are collected from various broadcast sports channels and cover 10 sport action classes such as diving, golfing, kicking, lifting, horse riding and so on. We follow the experiment settings in [150], [68] and exploit their action bank features [117] with around 30, 000 feature dimensions.

Figure 4.5 shows some example images of these datasets. First, we compare the proposed method with several standard feature selection methods including mutual information (MI) [76], [47], minimum redundancy maximum relevance (mRMR) [110], Laplacian score (LS) [60], reliefF [70], joint robust feature selection (RFS) [104] and JSDFS [145]. Figure 4.6 shows 20 selected features on some sample training images on three face datasets and one digit dataset. We observe that the selected features by MI, mRMR and LS have concentrated distribution, while those of RFS, JSDFS and our proposed JFS-LDL are distributed dispersedly. This observation is consistent with the characteristics of joint feature selection utilized in these methods [145], [104]. Compared with JSDFS, RFS and reliefF methods, the selected features by our method are distributed in the areas that hold more discriminative information, *e.g.*, in the face datasets these points are mostly around eyes, nose and mouth.

We then evaluate our method on classification task using different numbers of selected features. After learning the projection matrix P, the low-dimensional representation of a test image x_{ts} is found as $P^T x_{ts}$, and nearest neighbor classifier is used to predict its label. To challenge our method, we also simulate corruption in the USPS dataset by replacing 20% of randomly selected pixels of each image with pixel value 255. Figure 4.7 illustrates the recognition rates of our approach and the compared feature selection methods versus varying feature dimensions on aforementioned datasets. In these graphs, the horizontal axis shows the ratio of reduced dimension to the original dimension. In this experiment, we also compare our method with another filter-based greedy feature selection method called sparse discriminative feature selection (SDFS) [145]. As these graphs illustrate, JFS-LDL notably improves the recognition rate over all the competing methods, across all dimensions. JFS-LDL maintains a relatively stable performance across different dimensions, and as the number of selected features decreases, its advantage becomes more obvious. When the images contain considerable noise, the recognition rates of compared feature selection methods are severely degraded, but



(a) Extended YaleB

(b) PIE

(c) AR-*Unobscured*

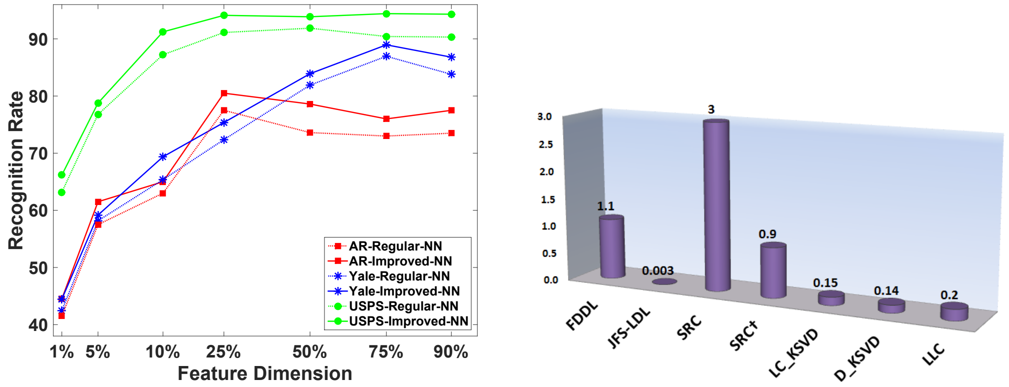
(d) AR-*Sunglasses+Scarf*

(e) Pixel Corrupted USPS

(f) UCF

Figure 4.7: Recognition rates (%) of different feature selection methods on various datasets

our approach can still obtain good results. For instance we have occlusion, illumination variations and facial expressions in AR-“Sunglasses+Scarf” scenario, and also simulated corruption in the USPS dataset. It is worth mentioning that the best performance of JFS-LDL is usually obtained while using a rela-



(a)

(b)

Figure 4.8: The performance of JFS-LDL method (a) Effect of considering sparse noise for NN classifier in JFS-LDL method on several datasets (b) Average classification time of an image using JFS-LDL method in the USPS dataset

tively small subset of features, *e.g.*, for the UCF action dataset the superior performance is achieved by exploiting just 10% of 30,000 features. These results indicate that our proposed feature selection method is able to select a robust and discriminative subset of features.

The recognition rate of NN classifier could be improved by taking the sparse error into account. The recovered sparse error of training samples could be assumed as the noise in the testing samples. Denote x_{ts} a test image, we perform the following operation:

$$\hat{x_{ts}} = x_{ts} - E_i \quad i = 1, \dots, N \tag{4.18}$$

If the sparse noise E_i is close to the noise of x_{ts}, then $\hat{x_{ts}}$ is cleaner and more robust than x_{ts} to be used for classification. So, we first obtain the low-dimensional representation of the test image as $P^T\hat{x_{ts}}$, and estimate its label as the majority of predicted labels by all training images $i = 1, \dots, N$. Figure 4.8a compares the recognition rates of using $P^T x_{ts}$ and $P^T\hat{x_{ts}}$, for three datasets including the AR-*Sunglasses+Scarf*, Extended YaleB and 20% pixel corrupted USPS, versus different dimensions. We observe that exploiting sparse noise generally improves the recognition rates, when the noise type is similar in training and test images; however, it would increase the classification time. Hence, we use the usual NN classifier (without aforementioned improvement)

Table 4.1: Recognition rates (%) of JFS-LDL using SVM on different datasets

	Lin-SVM				RBF-SVM			
Dataset	5%	25%	50%	75%	5%	25%	50%	75%
YaleB	85.4	96.3	97.2	97.5	90.0	95.9	96.8	97.5
AR$_{Un}$	70.2	78.5	90.3	92.5	78.4	89.5	96.4	97.5
AR$_{SS}$	68.2	74.5	83.3	84.5	74.4	80.5	88.4	89.5
PIE	97.7	98.5	99.6	99.9	98.1	99.0	99.6	99.9
USPS	73.4	90.4	90.9	91.8	83.9	94.3	95.6	96.9
UCF	95.2	96.1	97.2	98.0	95.4	96.2	99.0	96.4

in the following experiments.

We further evaluate the performance of JFS-LDL using linear and non-linear (with RBF kernel) SVM classifier. We use One-Against-All SVM for multi-class classification and SVM parameters are selected by cross-validation. Table 4.1 shows the average recognition rates on the above mentioned datasets. Here, the percentages show the ratio of reduced dimension to the original dimension, and AR$_{Un}$ and AR$_{SS}$ denote "Unobscured" and "Sunglasses+Scarf" scenarios, respectively. We observe that linear and non-linear SVM achieve higher accuracies than NN in all datasets, and this difference is more noticeable in the lower dimensions.

Table 4.2: Recognition rates (%) of JFS-LDL and some sparse learning methods on different datasets

Dataset	SRC	SRC†	D-KSVD	FDDL	LC-KSVD	LLC	JFS-LDL
	[140]	[140]	[160]	[150]	[68]	[133]	
YaleB	97.2	80.2	94.1	97.0	96.7	90.7	97.8 (**50%**)
AR$_{Un}$	97.5	66.5	88.8	92.7	93.7	88.7	98.5 (**30%**)
AR$_{SS}$	80.1	55.5	80.0	82.3	81.4	78.0	88.5 (**30%**)
PIE	93.0	90.2	89.3	97.0	91.8	90.3	99.9 (**30%**)
USPS	93.9	78.5	68.9	97.1	96.4	95.5	95.3 (**60%**)
UCF	92.9	83.6	88.1	94.3	91.2	87.5	99.1 (**5%**)

Besides, we compare the recognition rate of the proposed JFS-LDL with SRC and some of the recently proposed dictionary learning methods on five datasets in Table 4.2. We also evaluate the performance of SRC when using the same size as the dictionary, denoted as SRC†. In this experiment, we use non-linear SVM with RBF kernel as the classifier of our JFS-LDL. For each dataset, the projected dimension varies between 5% to 90% of the original dimension and the best achieved results among all dimensions, as well as the corresponding dimension are reported. Here, the number in the parenthesis show the fraction of the original feature, by which the optimal recognition rate is obtained. The results suggest that JFS-LDL with a selected subset of features is superior or competitive to other methods that are using all the features. This implies the effectiveness of our method in capturing the discriminative information for classification in the lower dimensions. The combination of LR-FDDL and joint optimization of the projection matrix, leads to more compactness within the same class and more dissimilarity between different classes. Consequently, we have well-separated classes in the reduced space and a simple (and fast) classifier such as NN or SVM performs very well. In contrast to the SRC and dictionary learning methods that use time-consuming l_1-minimization for classification, our proposed JFS-LDL can benefit from an efficient and fast classification schema. For instance, we illustrate the average classification time of a random test image on the USPS dataset in Figure 4.8b using a non-linear SVM classifier. We used a machine with 12GB RAM and Intel Core i7-3770 CPU. As it can be seen, the proposed JFS-LDL even with non-linear SVM as classifier, is much faster than SRC and other dictionary learning methods, and this is a desirable property for large-scale image classification tasks.

Finally, we design an experiment to verify the role of low-rank matrix recovery in the proposed feature selection method. Instead of LR-FDDL to find the dictionary and sparse coefficients, we utilize the original FDDL model, which uses original data vectors X_i as the input of model. Like before, we exploit NN and SVM classifiers to predict the labels. The results are presented in Table 4.3. It reflects that our approach can improve the recognition results over FDDL remarkably. We note that JFS with LR-FDDL noticeably outperforms JFS with FDDL in all evaluated datasets, and this difference is more significant in face datasets that have illumination and/or pose variations, facial expressions and occlusion.

4.5.3 Tumor Cell Counting

The accurate estimation of specific cells, is a determinative factor to have precise diagnosis in many medical scenarios. For instance, the number of proliferating tumor cells *e.g.*, Ki67 positive, is an important index associated with the severity of breast cancer disease. Traditional image processing techniques fail to provide a good estimation in these microscopic images, since the tumor cells are barely distinguishable from surrounding normal tissue like vessels, fat and fibrous tissue [42]. Some samples of these images are illustrated in Figure 4.9. So, we want to use the proposed JFS-LDL method for counting Ki67 positive cells in microscopic images.

As in medical applications there are very little training data available, we use excessive data augmentation. We tile each training image to generate numeros patches and also rotate the patches with differnt degrees. We treat these patches as the training data and associate each one with a label, indicating the number of target cells present in that patch. We may extract low-level hand-

Table 4.3: The role of low-rank matrix recovery component in JFS-LDL on different datasets

	JFS with FDDL			JFS with LR-FDDL		
Dataset	NN	Lin-SVM	RBF-SVM	NN	Lin-SVM	RBF-SVM
YaleB	35.9	79.2	83.1	87.3	98.2	97.8
AR$_{Un}$	56.2	73.5	76.3	81.5	99.9	98.5
AR$_{SS}$	50.2	68.5	70.3	75.5	84.9	89.5
PIE	80.8	91.5	94.6	99.9	99.9	99.9
USPS	88.4	83.4	88.1	94.8	91.4	95.3
UCF	81.2	94.1	95.6	98.0	99.0	99.1

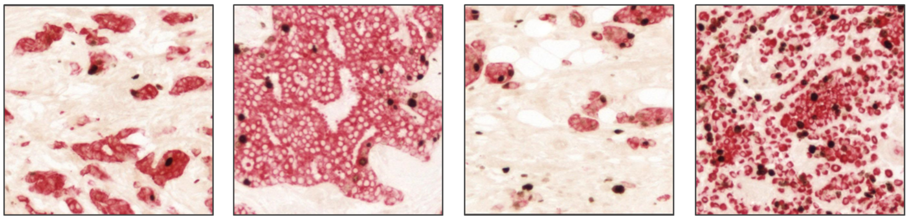


Figure 4.9: Example images from Ki67 tumor cell microscopic images dataset

crafted features such as LBP [3], SIFT [92] and HOG [31] from these patches. However, recent research has shown that we can benefit from the superior performance of deep neural networks in discovering multiple levels of representation and providing a high level and abstract representation of the data. So, we utilize CNN with the popular AlexNet architecture [75], and Euclidean loss function. In the training phase, augmented training data *i.e.*, collected patches, are fed to CNN and the feature vector from the last fully-connected layer is extracted for each patch. We use this 4096-dimensional feature as the input of our model *i.e.*, x_i. With all the x_is found by CNN, we will have the whole training samples as $X = [x_1, \ldots, x_N]$, where N is the total number of patches. It should be highlighted that we treat the cell counting as a classification problem, in which the cell count is regarded as an individual class, and patches with the same number of cells are considered as the same class. Having this in mind, we build the set of training samples as $X = [X_1, X_2, \ldots, X_K]$, where X_i denotes the training samples from the i^{th} class. We then perform Algorithm 4.1 to find the projection matrix P as described earlier.

In the test phase, the test image is tiled into several overlapping patches using a sliding-window technique. Each of these patches is fed into the learnt CNN model and the corresponding feature from the last fully-connected layer is extracted. We find the low-dimensional representation of the j^{th} patch as $P^T x_j$, and then use One-Against-All non-linear SVM with RBF kernel to predict its count (label). Once we predicted the cell counts for all the patches in one test image, we perform a 2-D linear interpolation over the estimated cell counts and corresponding coordinates to provide a spatial density prediction. By integrating these interpolated counts on pixel locations, we obtain the global count on each test image.

For the experiments, we use a dataset collected at the department of Laboratory Medicine in University of Alberta. This datasets contains 55 high-resolution (1920×2560) microscopic images, from which we use 45 images for training and the rest for test. The tumor cell size is about 10 to 20 pixels in diameter or 10 micrometer in physical length. On an average there are 2045 tumor cells per image, and this number varies in the range of $[70 - 4808]$. For the data augmentation step we set the patch size as 200×200 pixels, and use and rotation step 30°. The number of cells per patch varies between 0 and 99, $i.e., K = 100$.

To evaluate the counting accuracy on the testing images, we use Mean Absolute Error (MAE) and Mean Deviation Error (MDE) metrics as defined in Chapter 3. We compare our method with some counting methods including two shallow methods presented for cell counting [80] and cell detecting [7], and two deep models including the trained CNN which we utilized its learnt features and another deep network proposed for counting [118]. According to the results presented in Table 4.4, JFS-LDL has the least MAE and MDE on this dataset, and obtains the best performance. It is interesting to mention the best performance of JFS-LDL is achieved using just 60% of whole features (4096). Figure 4.10 demonstrates the estimated counts by JFS-LDL versus ground-truth for 10 test images using different feature dimensions. JFS-LDL closely follows the ground-truth in most cases.

Table 4.4: Error rate of JFS-LDL on Ki67 microscopic images dataset

Method	MAE	MDE
Deep Features [118]	189.35	0.12
Trained AlexNet on Ki67 dataset	151.20	0.09
Learning to Detect [7]	259.67	0.15
Learning to Count [80]	185.93	0.11
JFS-LDL	144.18	0.08

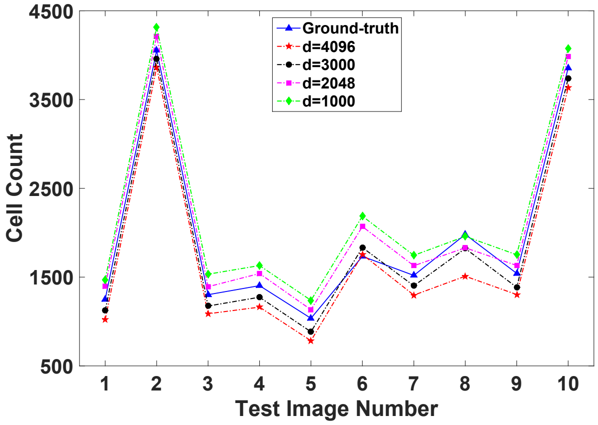


Figure 4.10: Estimated counts by JFS-LDL method versus ground-truth on Ki67 microscopic images dataset

4.6 Summary

In this chapter, we proposed a supervised feature selection method to identify the relevant feature subset from noisy high-dimensional features. When data are contaminated with severe noise such as occlusion, illumination and pose variations, the performance of most existing feature selection methods is unsatisfactory. We leverage the combination of low-rank matrix recovery and Fisher discrimination dictionary learning to learn discriminative, yet robust sparse representations form noisy data. The proposed JFS-LDL considers the interaction amongst features, and preserves the learnt sparse reconstructive relationship of the data in the subset of selected features, through introducing sparse scatter matrices. Extensive experiments on benchmark datasets verify the great performance of JFS-LDL for feature selection and classification. Moreover, we adopted JFS-LDL for counting the number of Ki67 positive cells, and significantly reduced the counting estimation error.

Chapter 5

Object Classification with Joint Projection and Low-rank Dictionary Learning

5.1 Introduction

Image classification based on visual content is a very challenging task, mainly because there is usually large amount of intra-class variability, arising from illumination and viewpoint variations, occlusion and corruption [22]. Numerous efforts have been made to counter the intra-class variability by manually designing low-level features for classification task. Representative examples are Gabor features and LBP [3] for texture and face classification, and SIFT [92] and HOG [31] features for object recognition. Although the hand-crafted low-level features achieve great success for some controlled scenarios, designing effective features for new data and tasks usually requires new domain knowledge since these features cannot be simply adopted to new conditions. Learning features from data itself is considered as a plausible way to overcome the limitations of low-level features [22], and successful examples of such learning methods are dictionary learning and deep learning.

The main idea of deep learning is to discover multiple levels of representation, with the hope that higher level features represent more abstract semantics of the data. Such abstract representations learned from a deep network are expected to provide more invariance to intra-class variability, if we train the deep model using a large amount of training samples [22]. One key ingredient for this success is the use of convolutional architectures that has shown remarkable performance at a number of different vision tasks including classification. In practice, we do not usually train an entire CNN from scratch with random initialization, because it is relatively rare to have a dataset of sufficient size. So, for small-sized training datasets, transfer learning is utilized as a powerful tool to enable training the target network. The usual approach is to replace and retrain the classifier on top of the CNN on the target dataset, and also fine-tune the weights of the pretrained network by continuing the backpropagation. However, the effectiveness of feature transfer declines when the base and target tasks become less similar [154]. Besides, when the target dataset is small, complex models like CNNs tend to overfit the data easily [154]. It could be even more complicated in classification tasks such as face recognition, in which the intra-class variability is often greater than the inter-class variability due to pose, expression and illumination changes and occlusion.

In contrast, the recent variations of dictionary learning methods have demonstrated great successes in image classification tasks on both small-sized and large intra-class variation datasets [44], [82]. To alleviate the effect of intra-class variations, low-rank constraint has been integrated into dictionary learning framework and impressive classification results have been reported [82], [161] for different kinds of contaminated observations such as occlusion, corruption, pose and lighting variations. Nevertheless, there is one key point that is ignored by low-rank dictionary learning methods. The fact is the dimensionality reduction and dictionary learning processes should be jointly conducted for a more effective classification, as discussed in Section 2.7. On the one hand, the existing joint dimensionality reduction and dictionary learning methods cannot handle noisy and large intra-class observations. One the other hand, current low-rank dictionary learning methods cannot select the best features on top of which dictionaries can be better learned, due to a separated dimensionality reduction process. In this chapter, we explore the dictionary leaning, low-rank and dimensionality reduction spaces simultaneously and propose an object classification method for noisy and large intra-class variation datasets, which have small-sized training set and may have high-dimensional

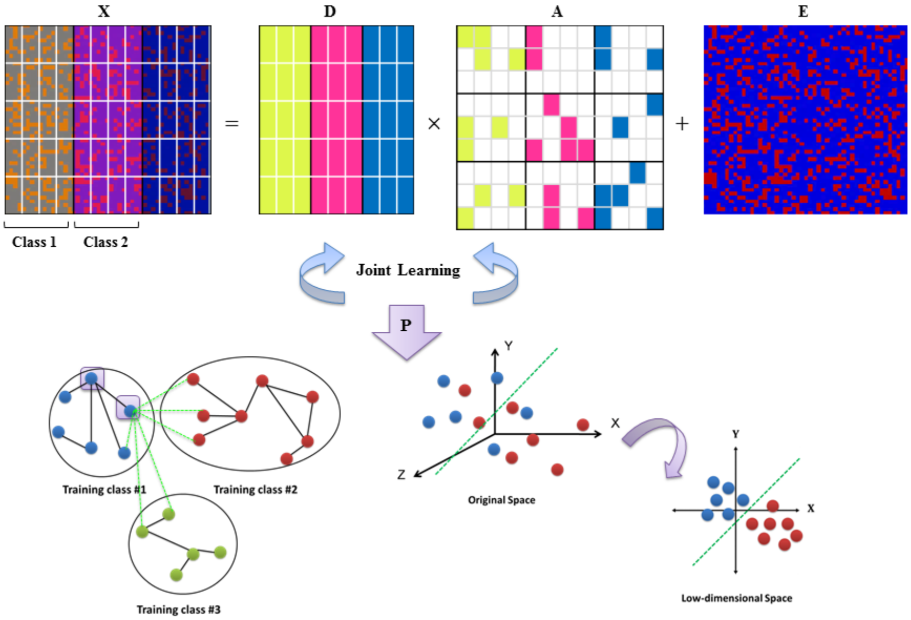


Figure 5.1: The overview of proposed JP-LRDL method

feature vectors. To the best of our knowledge, *this is the first proposed method that can handle all these issues simultaneously.*

To this end, we propose a novel framework called *Joint Projection and Low-rank Dictionary Learning using Dual Graph Constraints* (JP-LRDL). The basic idea of JP-LRDL is illustrated in Figure 5.1. The algorithm learns a discriminative structured dictionary in the reduced space, whose atoms have correspondences to the class labels and a graph constraint is imposed on the coding vectors to further enhance class discrimination. The *coefficient* graph makes the coding coefficients within the same class to be similar and the coefficients among different classes to be dissimilar. JP-LRDL specially introduces low-rank and incoherence promoting constraints on sub-dictionaries to make them more compact and robust to variations, and encourage them to be as independent as possible, respectively. Simultaneously, we consider optimizing the input feature space by jointly learning a subspace projection matrix. In particular, another graph is built on training data to explore intrinsic geometric structure of data. The *projection* graph enables us to preserve the desirable relationship among training samples and penalize the unfavorable relationships simultaneously. This joint framework empowers our algorithm with several important advantages: (1) Ability to handle large intra-class variation in observations, (2) Promoting the discriminative ability of the learned projection and dictionary, that enables us to deal with small-sized datasets, (3) Learning in the reduced dimensions with lower computational complexity, and (4) Maintaining both global and local structure of data.

5.2 The Proposed JP-LRDL Framework

We aim to learn a discriminative dictionary and a robust projection matrix simultaneously, using low-rank regularization and dual graph constraints. Let X be a set of m-dimensional training samples, *i.e.*, $X = [X_1, X_2, \ldots, X_K]$, where X_i denotes the training samples from class i and K is the number of classes. The structured class-specific dictionary is denoted by $D = [D_1, D_2, \ldots, D_K]$, where D_i is the sub-dictionary associated with class i. We also want to learn the projection (subspace learning) matrix $P \in R^{m \times d}\,(d < m)$, which projects data into a low-dimensional space. Denote by A the sparse representation matrix of the dimensionality reduced data $P^T X$ over dictionary D, *i.e.*, $P^T X \approx DA$. We can write A as $A = [A_1, A_2, \ldots, A_K]$, where A_i is the representation of $P^T X_i$ over D. Therefore, we propose JP-LRDL optimization model:

$$J_{(P,D,A)} = \underset{P,D,A}{\operatorname{argmin}} \left\{ R(P, D, A) + \lambda_1 \left\| A \right\|_1 + \lambda_2\, G(A) + \lambda_3 \sum_i \left\| D_i \right\|_* + \delta\, G(P) \right\} \tag{5.1}$$

$$s.t. \quad P^T P = I$$

where $R(P, D, A)$ is the discriminative reconstruction error, $\left\| A \right\|_1$ denotes the l$_1$-norm on coding coefficients, $G(A)$ is the graph-based coding coefficients, $\left\| D_i \right\|_*$ is the nuclear norm of sub-dictionary D_i , $G(P)$ represents the graph-based projection, and $\lambda_1, \lambda_2, \lambda_3, \delta$ are scalar parameters. We will discuss these terms in details.

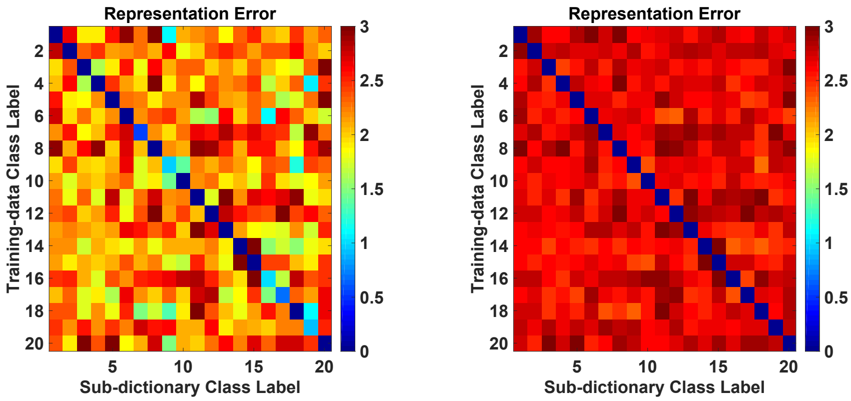
5.2.1 Discriminative Reconstruction Error Term

To learn a representative and discriminative structured dictionary, each sub-dictionary D_i should be able to well represent the dimensionality reduced samples from the i^{th} class, but not other classes. To illustrate this idea mathematically, we write A_i as $A_i = [A_i^1; \ldots; A_i^j; \ldots; A_i^K]$, where A_i^j is the representation coefficients of $P^T X_i$ over D_j. Our assumption implies that A_i^i should have significant coefficients such that $\|P^T X_i - D_i\, A_i^i\|_F^2$ is small, while for samples from class j $(j \neq i)$, A_i^j should have nearly zero coefficients, such that $\|D_j\, A_i^j\|_F^2$ is as small as possible. Moreover, the whole dictionary D should well represent dimensionality reduced samples from any class, which implies the minimization of $\|P^T X_i - D\, A_i\|_F^2$ in our model.

Furthermore, the common components of the samples in a dataset can be shared by a few or all the classes, especially in face datasets. Information redundancy in the original data leads to redundancy in the learned sub-dictionaries. So, in addition to the requirements of desirable discriminative reconstruction capability, we also need to promote incoherence among sub-dictionaries. We provide a structural incoherence constraint for sub-dictionaries as $\|D_i^T\, D_j\|_F^2$ for $i \neq j$. Thus, the discriminative reconstruction term is defined as:

$$\begin{aligned} R(P, D, A) = \sum_{i=1}^{K} \Big(&\|P^T X_i - D\, A_i\|_F^2 + \|P^T X_i - D_i\, A_i^i\|_F^2 \\ &+ \sum_{\substack{j=1 \\ j \neq i}}^{K} \|D_j\, A_i^j\|_F^2 + \sum_{\substack{j=1 \\ j \neq i}}^{K} \|D_i^T\, D_j\|_F^2 \Big) \end{aligned} \tag{5.2}$$

To better illustrate the role of the incoherence penalty term, we use a subset of the Caltech-101 object dataset [43]. This dataset is known for imaging variations such as scale, viewpoint, lighting and background. The subset includes 20 first classes with 20 training samples per class. We learn a dictionary by using the first three terms and all four terms of $R(P, D, A)$ and show the representation residuals of the training data over each sub-dictionary in Figures 5.2a and 5.2b, respectively. One can see that by using only the first three



(a) No incoherence penalty (b) With incoherence penalty

Figure 5.2: The role of the structural incoherence penalty term in JP-LRDL on a subset of the Caltech-101 dataset

terms of Equation (5.2), some training data may have large representation residuals over their associated sub-dictionaries because they can be partially represented by other sub-dictionaries. By adding incoherence term in Equation (5.2), D_i will have the minimal representation residual for X_i and the redundancy among sub-dictionaries would be reduced effectively.

5.2.2 Graph-based Coding Coefficient Term

To further increase the discrimination capability of dictionary D, we enforce the coding coefficient matrix A to be discriminative. Intuitively, discrimination can be assessed by the similarity of pairs of coding vectors from the same class, and the dissimilarity of pairs of coding vectors from the different classes. This can be achieved by constructing a *coefficient* graph, and maximizing the intra-class compactness and inter-class separability of coding coefficients through proper definition of graph weights. Denote the training data $X = \{x_1, x_2, \dots, x_N\}$, their corresponding sparse representations can be written as $A = \{\alpha_1, \alpha_2, \dots, \alpha_N\}$, where N is the number of training samples. We need to define an affinity matrix W^c for the *coefficient* graph, to measure the similarity of the sparse codes α_i and α_j according to their label and appearance.

As the first step, the matrix of the data samples in the i^{th} class, X_i, is decomposed into a low-rank matrix L_i and sparse noise E_i using low-rank matrix recovery [16] as follows:

$$\min_{L_i,E_i} \|L_i\|_* + \eta\|E_i\|_1 \quad s.t. \quad X_i = L_i + E_i \quad \forall i = 1,\ldots.K \tag{5.3}$$

Then, we define the weight matrix of the *coefficient* graph as follows:

$$W_{ij}^c = \begin{cases} 1 & \text{if } L(x_i) \in \mathcal{N}_{k_1}(L(x_j)) \text{ or } L(x_j) \in \mathcal{N}_{k_1}(L(x_i)) \text{ and } \ l(x_i) = l(x_j) \\ \\ -1 & \text{if } L(x_i) \in \mathcal{N}_{k_2}(L(x_j)) \text{ or } L(x_j) \in \mathcal{N}_{k_2}(L(x_i)) \text{ and } \ l(x_i) \neq l(x_j) \\ \\ 0 & \quad \text{otherwise} \end{cases} \tag{5.4}$$

where $L(x_i)$ is the corresponding low-rank representation of image x_i found by Equation (5.3), $\mathcal{N}_{k_1}(L(x_i))$ denotes the k_1 nearest neighbors of this representation and $l(x_i)$ is the label of image x_i. Utilizing the low-rank representation of images to determine their nearest neighbors enables us to constrain the intra-class coding coefficients to be similar, while the inter-class coefficients to be significantly dissimilar, even if the images are contaminated by nuisance factors.

It is reasonable to use the weighted sum of the squared distances of pairs of coding vectors as an indicator of the discrimination capability; so the discriminative coefficient term is defined as:

$$G(A) = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{1}{2}\|\alpha_i - \alpha_j\|_2^2 W_{ij}^c \tag{5.5}$$

This term ensures that the difference of the sparse codes of two images is minimized if they are from the same class and look similar, and the difference of the sparse codes of two images is maximized, if they are from different classes but also look similar. Equation (5.5) can be further simplified as:

$$G(A) = tr(A^T D^c A) - tr(A^T W^c A) = tr(A^T L^c A) \tag{5.6}$$

where D^c is a diagonal matrix of column sums of W^c as $D_{ii}^c = \sum_j W_{ij}^c$ and L^c is the Laplacian matrix as $L^c = D^c - W^c$. Interestingly, [14] showed that Fisher discrimination criterion, which is the most common discriminative coding coefficients term and originally adopted in [150], can be reformulated as a special case of the discrimination term in (5.5).

5.2.3 Low-rank Regularization

The training samples in each class are linearly correlated in many situations and reside in a low-dimensional subspace. So, the sub-dictionary D_i, which is representing data from the ith class, is reasonably low-rank. Imposing low-rank constraint on sub-dictionaries, makes them compact and mitigates the influence of noise and variations [82]. To find the most compact bases, we need to minimize $\|D_i\|_*$ for all classes in our optimization.

5.2.4 Graph-based Projection Term

We aim to learn a projection matrix that can preserve useful information and map the training samples to a discriminative space, where different classes are more discriminant toward each other, compared to the original space. Given this, we build the *projection* graph, using the training data matrix X and its corresponding class label set.

First, we find the low-rank representation of each image $x_i \in X$, using Equation (5.3). Then, the weight matrix W^p of the *projection* graph is defined as follows:

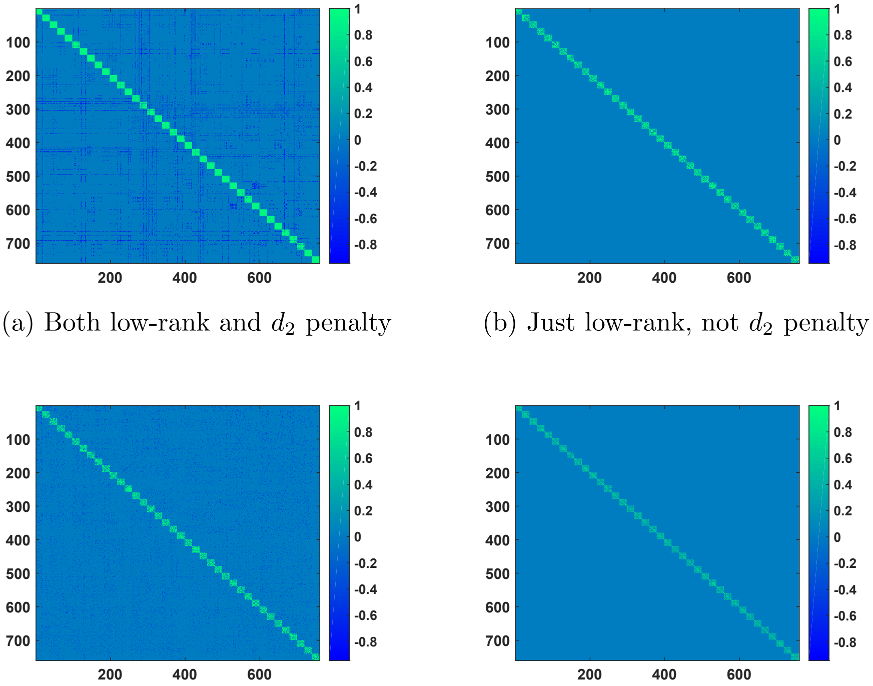
$$W_{ij}^p = \begin{cases} d_1 & \text{if } L(x_i) \in \mathcal{N}_{k_1}(L(x_j)) \text{ or } L(x_j) \in \mathcal{N}_{k_1}(L(x_i)) \text{ and } \quad l(x_i) = l(x_j) \\ \\ d_2 & \text{if } L(x_i) \in \mathcal{N}_{k_2}(L(x_j)) \text{ or } L(x_j) \in \mathcal{N}_{k_2}(L(x_i)) \text{ and } \quad l(x_i) \neq l(x_j) \\ \\ 0 & \quad \text{otherwise} \end{cases} \tag{5.7}$$

Similarly, $L(x_i)$ is the corresponding low-rank representation of image x_i found by (5.3), $\mathcal{N}_{k_1}(L(x_i))$ denotes the k_1 nearest neighbors of this representation and $l(x_i)$ is the label of x_i.

To preserve the local geometrical structure in the projected space, one may naturally hope that, if two data points x_i and x_j are close in the intrinsic manifold, their corresponding low-dimensional embeddings y_i and y_j should also be close to each other. Here, the low-dimensional representative of an image x_i is obtained as $y_i = P^T x_i$. Ideally, similar data pairs which belong to different classes, in the original space should be far apart in the embedded space, and the affinity matrix defined in (5.7) accomplishes it. As the first advantage of the *projection* graph, it would enable us to preserve desirable relationship among training samples, and penalize unfavorable relationship among them at the same time. This is achieved by defining the weights $d_1(x_i, x_j) = exp(-\|L(x_i) - L(x_j)\|^2/2t^2)$ and $d_2(x_i, x_j) = -exp(-\|L(x_i) - L(x_j)\|^2/2t^2)$, where t is considered as 1 here. More importantly, these relationships should be persevered or penalized even if the images are heavily corrupted, occluded or pose/illumination varied. Accordingly, we exploit the low-rank representation of images to determine their nearest neighbors and also to assign the weights of the matrix, rather than their original representations.

We design an experiment to verify the importance of contributing components of the *projection* graph. Here, we illustrate the weight matrix W^p for the Extended YaleB [54] face dataset, which is known for different illumination conditions, and for extra challenge we also simulate corruption by replacing 60% of randomly selected pixels of each image with pixel value 255. There are 38 subjects in the dataset, and we randomly select 20 training samples per class. Ideally the connecting weights between similar images from the same and different classes should be large and small, respectively. If we promote the former and penalize the latter, we would be able to keep these relationships in the low-dimensional space as well. Figure 5.3a shows the weight matrix found by (5.7), which is confirming to the ideal case.

There are two contributing factors in building this weight matrix, that we need to verify their importance. First, to spot the role of low-rank, we recalculate the weight matrix, without utilizing low-rank representation (neither in neighborhood determination, nor in weights assignment) and demonstrate it in Figure 5.3c. Clearly, if we ignore the low-rank representation of images, corruption and illumination variations significantly deteriorates the weight matrix. Then, to verify the importance of penalizing unfavorable relations among similar training samples from different classes, we ignore the second condition of (5.7) and simply set all weights to zero; except those between similar pairs from the same class, which is obtained by $d_1(x_i, x_j)$. Figures 5.3b and 5.3d



(a) Both low-rank and d_2 penalty

(b) Just low-rank, not d_2 penalty

(c) Just d_2 penalty, not low-rank

(d) Neither low-rank nor d_2 penalty

Figure 5.3: Comparison of different possible variations of weight matrix W^p for 60% pixel-corrupted images on the Extended YaleB dataset

show these weight matrices with and without exploiting low-rank representation, respectively. Compared to these matrices, our weight assignment schema as shown in Figure 5.3a, is much more discriminative, robust to variations, and more similar to the ideal case.

All things considered, we formulate the graph-based projection $G(P)$ term as follows:

$$G(P) = \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{1}{2}\|y_i - y_j\|_2^2\, W_{ij}^p \tag{5.8}$$

Let D^p be a diagonal matrix of column sums of W^p, $D_{ii}^p = \sum_j W_{ij}^p$ and L^p the Laplacian matrix as $L^p = D^p - W^p$. The cost function in Equation (5.8) can be reduced to:

$$G(P) = tr(P^T X L^p X^T P) \quad s.t. \quad P^T X D^p X^T P = I \tag{5.9}$$

We note that the constraint $P^T X D^p X^T P = I$ removes the arbitrary scaling factor in the embedding. In order to make the constraint simpler, here we use the normalized graph Laplacian [29] as $\hat{L}^p = I - D^{p^{-\frac{1}{2}}} W^p D^{p^{-\frac{1}{2}}}$. Consequently, Equation (5.9) is reformulated as:

$$G(P) = tr(P^T X \hat{L}^p X^T P) \quad s.t. \quad P^T P = I \tag{5.10}$$

By incorporating Equations (5.2), (5.6) and (5.10) into the main optimization model, the JP-LRDL model is built as:

$$\begin{aligned} &\min_{P,D,A} \sum_{i=1}^{K} \Big(\|P^T X_i - D\, A_i\|_F^2 + \|P^T X_i - D_i\, A_i^i\|_F^2 + \sum_{\substack{j=1 \\ j \neq i}}^{K} \|D_j\, A_i^j\|_F^2 + \sum_{\substack{j=1 \\ j \neq i}}^{K} \|D_i^T\, D_j\|_F^2 \Big) \\ &+ \lambda_1 \|A\|_1 + \lambda_2\, tr(A^T L^c A) + \lambda_3 \sum_{i=1}^{K} \|D_i\|_* + \delta\, tr(P^T X \hat{L}^p X^T P) \quad s.t. \quad P^T P = I \end{aligned} \tag{5.11}$$

5.3 Optimization

The objective function in Equation (5.11) can be divided into three sub-problems to jointly learn dictionary D, projection P and coding coefficients A. These sub-problems are optimized alternatively by updating one variable and fixing the other ones, through an iterative process. The outline of the proposed JP-LRDL is summarized in Algorithm 5.2 and each sub-problem is discussed in details in the following subsections.

5.3.1 Update of Coding Coefficients A

Assuming that D and P are fixed, the objective function in Equation (5.11) is further reduced to:

$$\begin{aligned} J_{(A)} = \underset{A}{\operatorname{argmin}} &\sum_{i=1}^{K} \Big(\|P^T X_i - D\, A_i\|_F^2 + \|P^T X_i - D_i\, A_i^i\|_F^2 + \sum_{\substack{j=1 \\ j \neq i}}^{K} \|D_j\, A_i^j\|_F^2 \Big) \\ &+ \lambda_1 \|A\|_1 + \lambda_2\, tr(A^T L^c A) \end{aligned} \tag{5.12}$$

We optimize A_i class-by-class and meanwhile, make all other $A_j(j \neq i)$ fixed. As a result, Equation (5.12) is simplified as:

$$J_{(A_i)} = \underset{A_i}{\operatorname{argmin}} \|P^T X_i - D\, A_i\|_F^2 + \|P^T X_i - D_i\, A_i^i\|_F^2 + \sum_{\substack{j=1 \\ j\neq i}}^{K} \big\|D_j A_i^j\big\|_F^2 \quad (5.13)$$
$$+\lambda_1 \left\|A_i\right\|_1 + \lambda_2\, tr(A_i^T L^c A_i)$$

Following the work in [77], we update A_i one by one in the i^{th} class. We define $\alpha_{i,p}$ as the coding coefficient of the p^{th} sample in the i^{th} class and optimize each $\alpha_{i,p}$ in A_i alternatively, by fixing the coding coefficients $\alpha_{j,p}(j \neq i)$ for other samples, and rewrite (5.13) as:

$$J_{(\alpha_{i,p})} = \underset{\alpha_{i,p}}{\operatorname{argmin}} \|P^T X_i - D\, \alpha_{i,p}\|_F^2 + \|P^T X_i - D_i\, \alpha_{i,p}^i\|_F^2 + \sum_{\substack{j=1 \\ j\neq i}}^{K} \|D_j \alpha_{i,p}^j\|_F^2$$
$$(5.14)$$
$$+\lambda_1 \left\|\alpha_{i,p}\right\|_1 + \lambda_2\, \mathcal{Q}(\alpha_{i,p})$$

where

$$\mathcal{Q}(\alpha_{i,p}) = \lambda_2 \Big(\alpha_{i,p}^T A_i L_p^c + (A_i L_p^c)^T \alpha_{i,p} - \alpha_{i,p}^T L_{pp}^c \alpha_{i,p} \Big) \quad (5.15)$$

where L_p^c is the p^{th} column of L^c, and L_{pp}^c is the entry in the p^{th} row and p^{th} column of L^c. We then apply the feature-sign search algorithm [77] to solve $\alpha_{i,p}$ in (5.14).

5.3.2 Update of Dictionary D

Next, we optimize D while A and P are fixed. We update D_i class-by-class, by fixing all other sub-dictionaries $D_j(j \neq i)$. By ignoring irrelevant terms, the objective function (5.11) reduces to:

$$J_{(D_i, A_i^i)} = \underset{D_i, A_i^i}{\operatorname{argmin}} \Big\{ \|P^T X_i - D_i\, A_i^i - \sum_{\substack{j=1 \\ j\neq i}}^{K} D_j\, A_i^j\|_F^2 + \|P^T X_i - D_i\, A_i^i\|_F^2$$
$$(5.16)$$
$$+ \sum_{\substack{j=1 \\ j\neq i}}^{K} \|D_j\, A_i^j\|_F^2 + \sum_{\substack{j=1 \\ j\neq i}}^{K} \|D_i^T\, D_j\|_F^2 + \lambda_3 \|D_i\|_* \Big\}$$

To solve Equation (5.16), we first define a sub-dictionary fidelity term $r(D_i)$ as:

$$r(D_i) = \|P^T X_i - D_i\, A_i^i - \sum_{\substack{j=1\\ j\neq i}}^{K} D_j\, A_i^j\|_F^2 \; + \sum_{\substack{j=1\\ j\neq i}}^{K} \|D_j\, A_i^j\|_F^2 + \sum_{\substack{j=1\\ j\neq i}}^{K} \|D_i^T\, D_j\|_F^2 \tag{5.17}$$

When D_i is updated, the coding coefficients of $P^T X_i$ over D_i, *i.e.*, A_i^i should also be updated to reflect this change. We enforce sparsity on $\|A_i^i\|_1$ to both avoid the trivial solution, and keep sparsity constraint on coding coefficients. We also introduce a sparse error term E_i here to alleviate the effect of noise, *i.e.*, $P^T X_i = D_i\, A_i^i + E_i$. We adopt $\|E_i\|_{2,1}$ to characterize the error, since we want to model the sample-specific corruption and outliers, and it has been shown [87] that $\|.\|_{2,1}$ is preferred to $\|.\|_F$ and $\|.\|_1$ for handling sample-specific corruptions. The $\mathrm{l}_{2,1}$-norm encourages the columns of E to be zero, which is consistent with our assumption in the paper, that some vectors in data are corrupted. Therefore, Equation (5.16), can be converted to the following form:

$$\begin{aligned} &\min_{D_i, A_i^i, E_i} \|A_i^i\|_1 + \lambda_3\, \|D_i\|_* + \beta\, \|E_i\|_{2,1} + \lambda\, r(D_i) \\ &\qquad s.t. \quad P^T X_i = D_i\, A_i^i + E_i \end{aligned} \tag{5.18}$$

To facilitate the optimization, we introduce two relaxation variables J and Z; so Equation (5.18) is rewritten as:

$$\begin{aligned} &\min_{D_i, A_i^i, E_i} \|Z\|_1 + \lambda_3\, \|J\|_* + \beta\, \|E_i\|_{2,1} + \lambda\, r(D_i) \\ &s.t. \quad P^T X_i = D_i\, A_i^i + E_i,\; D_i = J,\; A_i^i = Z \end{aligned} \tag{5.19}$$

The above problem can be solved by inexact ALM [86]. The augmented Lagrangian function of (5.19) is:

$$\begin{aligned} &\min_{D_i, A_i^i, E_i} \|Z\|_1 + \lambda_3\, \|J\|_* + \beta\, \|E_i\|_{2,1} + \lambda\, r(D_i) \\ &+ tr\big[T_1^T (P^T X_i - D_i\, A_i^i - E_i)\big] + tr\big[T_2^T (D_i - J)\big] + tr\big[T_3^T (A_i^i - Z)\big] \\ &+ \frac{\mu}{2}\Big(\|P^T X_i - D_i\, A_i^i - E_i\|_F^2 + \|D_i - J\|_F^2 + \|A_i^i - Z\|_F^2\Big) \end{aligned} \tag{5.20}$$

where T_1,T_2 and T_3 are Lagrange multipliers, and μ is a balance parameter indicating the step size. The details of solving Equation (5.20) can be found in Algorithm 5.1. We provide some more details about the solutions of this algorithm:

- Step 3 is solved by soft-thresholding (shrinkage) operator as defined in Equation (2.16). In general $\Omega_\varepsilon[W] = \underset{X}{\operatorname{argmin}}\ \varepsilon\|X\|_1 + \frac{1}{2}\|X - W\|_F^2$.
- Step 5 is solved by singular value thresholding operator, which consists of SVD and thresholding as: $U\,\Omega_\varepsilon[S]\,V^T = \underset{X}{\operatorname{argmin}}\ \varepsilon\|X\|_* + \frac{1}{2}\|X - W\|_F^2$.
- Step 6 is in the form of $AX + XB = C$, which is a standard Sylvester equation, and can be effectively solved using the existing tools [126], [56].
- For step 7, the optimal solution of $\min_X\ \lambda\|X\|_{2,1} + \frac{1}{2}\|X - Q\|_F^2$ is X^*, such that the i^{th} column of X^* is obtsined as [88]:

$$X^*(:,i) = \begin{cases} \dfrac{\|q_i\| - \lambda}{\|q_i\|}\,\|q_i\| & if\ \lambda < \|q_i\| \\ \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

where q_i is the i^{th} column of Q.

5.3.3 Update of Projection Matrix P

In order to solve for P, we keep D and A fixed. As a result, the objective function in Equation (5.11) is then reduced to:

$$J_{(P)} = \underset{P}{\operatorname{argmin}} \left\{ \sum_{i=1}^{K} \left(\|P^T X_i - D\,A_i^i\|_F^2 + \|P^T X_i - D_i\,A_i^i\|_F^2 \right) + \delta\, tr(P^T X \hat{L}^p X^T P) \right\}$$
$$s.t. \quad P^T P = I \tag{5.22}$$

First, we rewrite the objective function in a more convenient form:

$$J_{(P)} = \underset{P}{\operatorname{argmin}} \left\{ \|P^T X - \hat{D}\hat{Z}\|_F^2 + \delta\, tr(P^T X \hat{L}^p X^T P) \right\} \quad s.t. \quad P^T P = I \tag{5.23}$$

where $\hat{D} = \Big[[D, D_1], [D, D_2], \ldots, [D, D_K] \Big]$, and $\hat{Z}$ is a block-diagonal matrix, whose diagonal elements are formed as $\hat{Z}_{ii} = [A_i\,;\, A_i^i]\ ;\ \forall i$. Because of the

Algorithm 5.1 Inexact ALM Algorithm for Equation (5.20)

```
Input: Low-dimensional data $P^T X_i$, Sub-dictionary $D_i$, Parameters $\lambda_3, \beta, \lambda$
Output: $D_i, E_i, A_i^i$
Initialize: $J = 0$, $E_i = 0$, $T_1 = 0$, $T_2 = 0$, $T_3 = 0$, $\mu = 10^{-6}$, $max_\mu = 10^{30}$, $\epsilon = 10^{-8}$, $\rho = 1.1$
while not converged do
    Fix other variables and update $Z$ as:
        $Z = \underset{Z}{\text{argmin}} \left(\frac{1}{\mu}\|Z\|_1 + \frac{1}{2}\|Z - (A_i^i + \frac{T3}{\mu})\|_F^2\right)$
    Fix other variables and update $A_i^i$ as:
        $A_i^i = \left(D_i^T D_i + I\right)^{-1}\left(D_i^T (P^T X_i - E_i) + Z + \frac{D_i^T T_1 - T_3}{\mu}\right)$
    Fix other variables and update $J$ as:
        $J = \underset{J}{\text{argmin}} \left(\frac{\lambda_3}{\mu}\|J\|_* + \frac{1}{2}\|J - (D_i + \frac{T2}{\mu}\|_F^2\right)$
    Length normalization for each column in $J$
    Fix other variables and update $D_i$ as:
        $\left(\frac{2\lambda}{\mu} \sum_{\substack{j=1 \\ j \neq i}}^{K} D_j D_j^T\right) D_i + D_i \left(\frac{2\lambda}{\mu} A_i^i A_i^{i^T} + A_i^i A_i^{i^T} + I\right)$
        $= \frac{2\lambda}{\mu}\left(P^T X_i A_i^{i^T} - \sum_{\substack{j=1 \\ j \neq i}}^{K} D_j A_i^j A_i^{i^T}\right) + P^T X_i A_i^{i^T} - E_i A_i^{i^T} + J + \frac{T_1 A_i^{i^T} - T_2}{\mu}$
    Length normalization for each column in $D_i$
    Fix other variables and update $E_i$ as:
        $E_i = \underset{E_i}{\text{argmin}} \left(\frac{\beta}{\mu}\|E_i\|_{2,1} + \frac{1}{2}\|E_i - (P^T X_i - D_i A_i^i + \frac{T1}{\mu})\|_F^2\right)$
    Update $T_1, T_2, T_3$ as:
        $T_1 = T_1 + \mu(P^T X_i - D_i A_i^i - E_i)$
        $T_2 = T_2 + \mu(D_i - J)$
        $T_3 = T_3 + \mu(A_i^i - Z)$
    Update $\mu$ as: $\mu = min(\rho\mu, max_\mu)$
    Check stopping conditions as:
  $\|D_i - J\|_\infty < \epsilon$  and  $\|P^T X_i - D_i A_i^i - E_i\|_\infty < \epsilon$  and  $\|A_i^i - Z\|_\infty < \epsilon$
end while
```

orthogonal constraint $P^T P = I$, we have $\|P^T X - \hat{D}\hat{Z}\|_F^2 = tr(P^T \varphi(P) P)$, where $\varphi(P) = \big(X - P\hat{D}\hat{Z}\big)\big(X - P\hat{D}\hat{Z}\big)^T$. So, Equation (5.22) is reformulated as:

$$J_{(P)} = \underset{P}{\text{argmin}}\; tr\Big(P^T\big(\varphi(P) + \delta(X\hat{L}^p X^T)\big)P\Big) \quad s.t. \quad P^T P = I \tag{5.24}$$

To solve the above minimization, we iteratively update P according to the projection matrix obtained in the previous iteration. Using singular value decomposition we have $[U, \Sigma, V^*] = SVD\Big(\varphi(P) + \delta\,(X\hat{L}^p X^T)\Big)$. Then, we can update P as the eigenvectors in U associated with the first c smallest eigenvalues of Σ, $i.e., P_t = U(1:d,:)$, where P_t is the projection matrix in the t^{th} iteration. To avoid big changes in P and make the optimization stable, we choose to update P gradually in each iteration as following:

$$P_t = P_{t-1} + \gamma\Big(U(1:c,:) - P_{t-1})\Big) \tag{5.25}$$

γ is a small positive constant to control the change of P in consecutive iterations.

Algorithm 5.2 JP-LRDL Algorithm

```
1: Initialize:
   Projection $P$ as LPP [62] of $X$
   Dictionary $D$; set the atoms of $D_i$ as the eigenvectors of $P^T X_i$
2: Update the coding coefficient matrix $A$
   Fix $D, P$ and solve $A_i i = 1, 2, \dots, K$; one by one by solving Equation (5.14) using Feature-sign search algorithm [77].
3: Update the dictionary $D$
   Fix $A, P$ and solve $D_i i = 1, 2, \dots, K$; one by one by solving Equation (5.20) using inexact ALM by Algorithm 5.1.
4: Update the projection $P$
   Fix $D, A$ and solve $P$ by solving Equation (5.25).
5: Output:
   Return to step 2 until the objective function values in consecutive iterations are close enough or the maximum number of iterations is reached. Then output $P$,$D$ and $A$.
```

5.4 Complexity and Convergence Analysis

5.4.1 Time Complexity

We analyze the time complexity of three sub-problems of JP-LRDL optimization as follows:

- To update the sparse coding coefficients, we exploit feature-sign search algorithm [77] with a time complexity of $O(sC)$, where s is the sparsity level of the optimal solution *i.e.,* the number of nonzero coefficients, and C is the dictionary size.

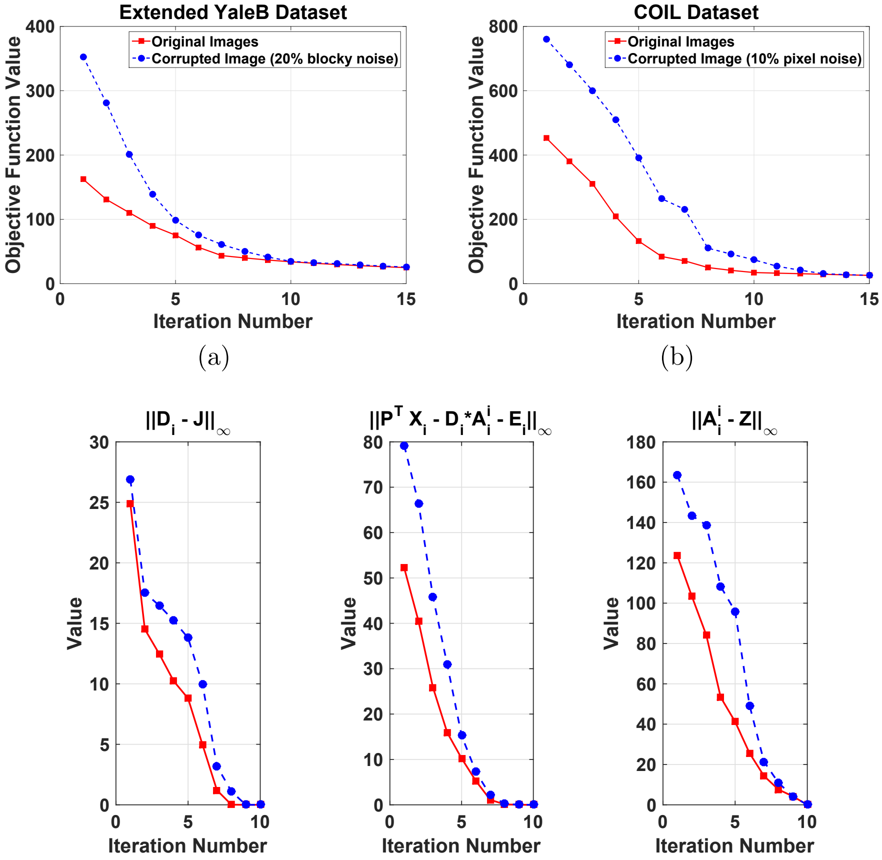
- For updating the dictionary we use Algorithm 5.1, in which steps 1 and 3 are the most time-consuming ones due to SVD with cost of $O(N^3)$, where N is the number of training samples. The matrix inverse calculation in step 4 costs $O(p_i^3)$, and the state-of-the-art solution to our Sylvester equation costs $O(d^3 + p_i^3)$, where d is the number of features in the subspace and p_i is the number of atoms of the sub-dictionary D_i. In all, the total complexity of Algorithm 5.1 is $t_1 \, O(N^3)$, where t_1 is the number of iterations of this algorithm.

- In the updating process of projection matrix, the most time-consuming step is SVD again, and the time complexity of this step would be $O(N^3)$.

Hence, the total time complexity of JP-LRDL is $t_2 \, O(N^3)$, where t_2 is the total number of iterations of Algorithm 5.2.

5.4.2 Convergence Analysis

Although Equation (5.11) is non-convex, the convergence of each sub-problem is guaranteed. For updating coding coefficients, we exploit feature-sign search algorithm, which [77] proved this algorithm converges to a global optimum of the optimization problem in a finite number of steps. For updating sub-dictionaries, we use inexact ALM as demonstrated in Algorithm 5.1. The convergence of inexact ALM with at most two blocks has been well studied and a proof to demonstrate its convergence property can be found in [86]. Liu *et al.* [87] also showed that there actually exist some guarantees for ensuring the convergence of inexact ALM with three or more blocks (here Z, J and E) under some mild consitions, and it could be well expected that Algorithm 5.1 has good convergence properties. Moreover, inexact ALM is known to generally



(a)

(b)

(c)

Figure 5.4: An example of JP-LRDL convergence on (a) the Extended YaleB dataset and (b) the COIL dataset (c) The curves of stopping conditions of Algorithm 5.1 on the Extended YaleB dataset versus the iteration number

perform well in reality, as illustrated in [162]. The convergence of updating the projection matrix in (5.25) has also been discussed in [44].

In addition, we demonstrate the convergence properties of our algorithm in practice. To verify the convergence for both clean and contaminated data, we simulate corruption and occlusion in this experiment. The images of the Extended YaleB face dataset [54] are manually corrupted by an unrelated block image at a random location and the percentage of corrupted area is considered as 20% of the image. For the COIL object dataset [101], we replace 10% of randomly selected pixels of each image with pixel value 255. Figures 5.4a and 5.4b illustrate the convergence curves of the proposed JP-LRDL on these datasets. It can be observed that JP-LRDL converges efficiently, and after several iterations the value of objective function becomes stable, such that local solutions cannot make the problem unpredictable. Although the objective function value on corrupted and occluded images is larger than that of original ones, the function converges very well after some iterations in both cases. Additionally, Figure 5.4c demonstrates the values of $\|D_i - J\|_\infty$, $\|P^T X_i - D_i\, A_i^i - E_i\|_\infty$ and $\|A_i^i - Z\|_\infty$, which are the stopping conditions of Algorithm 5.1, on the original and noisy images of the Extended YaleB dataset. We observe that inexact ALM efficiently convergences through few iterations in both cases.

We also evaluate the running time of JP-LRDL and other competing methods, that will be introduced in the following section, on the Extended YaleB dataset in Figure 5.5d. The training time is computed as the average over the entire training set, while fixing the projected dimension as 30% of the original dimension. We used a machine with 12GB RAM and Intel Core i7-3770 CPU. We observe that our method has a reasonable training time compared to the competing methods.

5.5 The Classification Scheme

Once D and P are learned, they could be used to represent a query sample x_{ts} and predict its label. First the test sample is projected into the low-dimensional space using P and then coded over D to find the sparse coding coefficients. Since the number of training samples of each class is relatively small, the learned sub-dictionary D_i may not be able to faithfully represent the testing samples of this class, and hence we code the testing sample over the whole dictionary D, by solving the following equation:

$$\hat{a} = \operatorname*{argmin}_a \left\{ \|P^T x_{ts} - Da\|_2^2 + \xi \|a\|_1 \right\} \tag{5.26}$$

where ξ is a positive scalar. The coding vector $\hat{a}$ can be written as $\hat{a} = [\hat{a}_1, \hat{a}_2, \dots \hat{a}_K]$ where $\hat{a}_i$ is the coefficient sub-vector associated with sub-dictionary D_i. The representation residual for the i^{th} class is calculated as:

$$e_i = \|P^T x_{ts} - D_i\, \hat{a}_i\|_2^2 + \omega \|\hat{a}_i - m_i\|_2^2 \tag{5.27}$$

where ω is a preset balancing weight and m_i is the learned mean vector of A_i. Incorporating the term $\|\hat{a}_i - m_i\|_2^2$ is to make the best of the discrimination within the dictionary, because the dictionary is learned to make coding coefficients similar from the same class and dissimilar among different classes. Finally, the identity of testing sample is determined by $label(x_{ts}) = \text{argmin}\ _i\{e_i\}$.

Similar to Subsection 4.5.2, the recognition rate of l_1-minimizer classifier can be improved by considering the sparse error. If we assume the recovered sparse error of training samples is similar to the noise in the testing samples, then for a test image x_{ts}, we remove the potential sparse noise of that as $\hat{x_{ts}} = x_{ts} - E_i \quad i = 1, \ldots, N$ and perform the classification using $\hat{x_{ts}}$, instead of x_{ts}. The label is then found as the majority of predicted labels by images $i = 1, \ldots, N$. This post-processing step could enhance the recognition rate, especially if the kind of variation or noise in the training and test images are similar. Since this step increases the classification time, we stick on the aforementioned classification method in (5.27), in the following experiments.

5.6 Experimental Results

The performance of JP-LRDL method is evaluated on various classification tasks. We compare our method with several related methods. FDDL [150] and $D^2L^2R^2$ [82] are representatives of conventional dictionary learning and low-rank dictionary learning methods, respectively. We also compare JP-LRDL with joint dimensionality reduction and dictionary learning methods including JNPDL [91], SDRDL [147], SE [103], LGE-KSVD [111] and JDDRDL [44]. Since SE can obtain at most K (number of classes) features in the reduced space, it is excluded from the experiments which the projected dimension is larger than K.

We evaluate the performance of our JP-LRDL and related methods on different face, object and action datasets. For constructing the training set, we select images randomly, and the random selection process is repeated 10 times and we report the average recognition rates for all methods. We set the

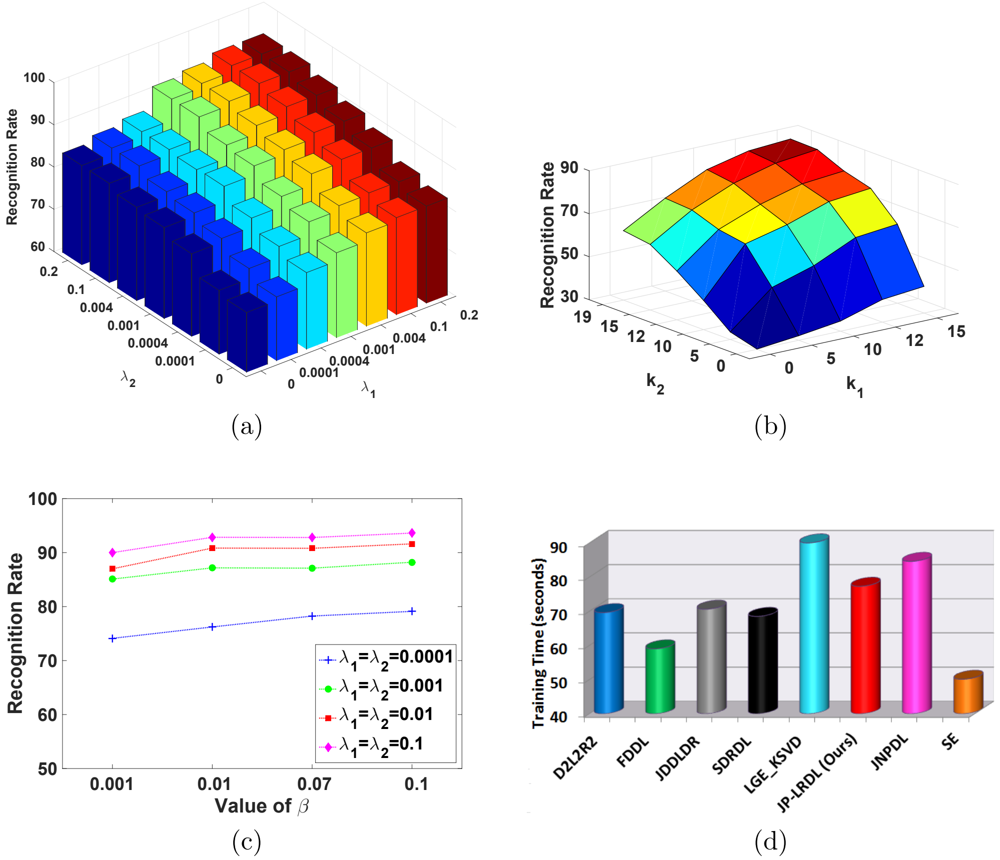


Figure 5.5: Recognition rate (%) of JP-LRDL with various parameter settings of (a) (λ_1, λ_2) parameters on the AR dataset (b) neighborhood size (k_1, k_2) on the Extended YaleB dataset (c) β parameter on the AR dataset and (d) The average training time of different methods on the Extended YaleB dataset

number of dictionary atoms of each class as training size. Also, we set the maximum iteration of all the iterative methods as 10. For all the competing methods, we use their original settings and all the hyper-parameters are found by 5-fold cross validation.

5.6.1 Parameters Selection

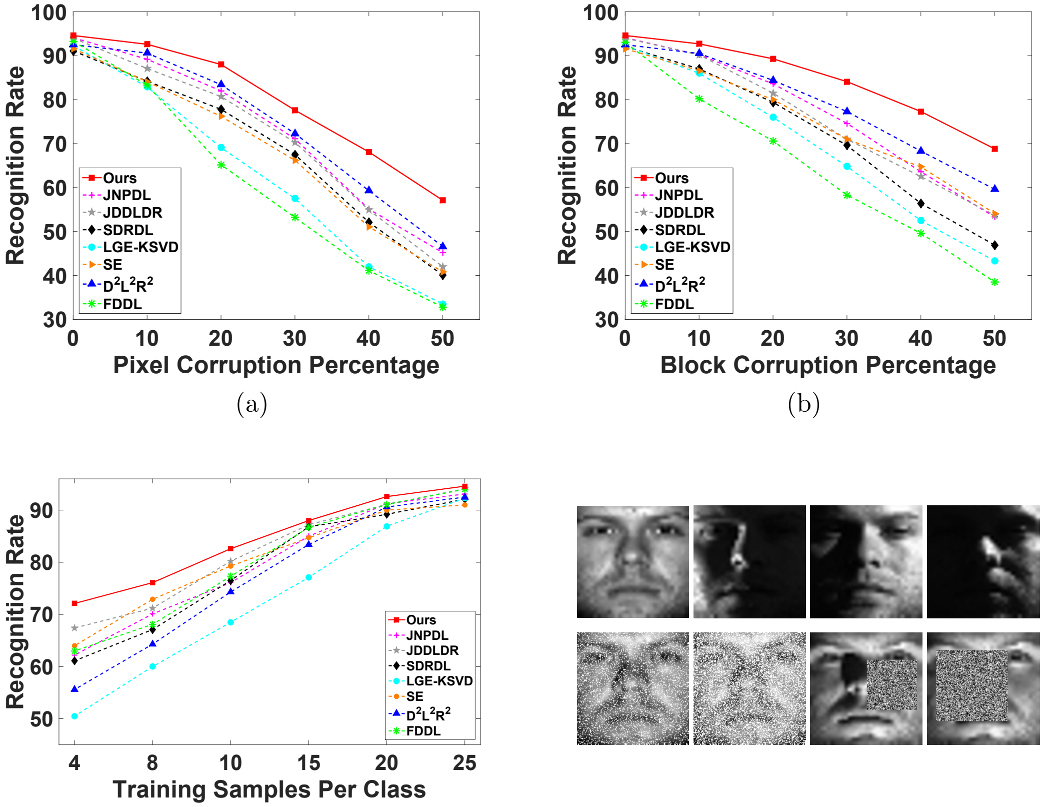
There are nine parameters in our model, which need to be tuned: λ_1, λ_2, λ_3, δ in Equation (5.11), β, λ in Equation (5.18), γ in Equation (5.25) and neighborhood parameters k_1, k_2 in graph-based terms. The first seven tuning parameters of JP-LRDL are chosen by 5-fold cross validation. However, we found out that changing λ_3, δ and λ parameters does not affect the results that much and we set them as 1. Because there are many combinations of remaining four parameters, we first search for the optimal value of λ_1, λ_2 between 0.0001 and 0.1, by fixing other parameters.

To investigate how sensitive the λ_1, λ_2 parameters are, we set the value of $\beta = 0.1, \gamma = 0.1$ and then explore the effects of the other two parameters. Figure 5.5a shows the recognition rate versus different values of these two parameters by fixing β, γ as 0.1 on the AR face dataset ("Sunglasses+Scarf" scenario as already explained in Subsection 4.5.2). We observe the accuracy reaches a plateau as either λ_1 or λ_2 grow from 0.1, and this trend is mostly similar in all evaluated datasets. We notice that when $\lambda_1 = 0$, the accuracy drops remarkably, which shows the importance of the sparsity of the coefficients. Figure 5.5c also illustrates the recognition rate versus the value of β, under four different pair values of λ_1, λ_2 on the AR face dataset. We note that JP-LRDL performs well in a reasonable range of β parameter and the highest accuracy belongs to the $\lambda_1 = \lambda_2 = 0.1$, which is consistent with the results from Figure 5.5a. We set the γ parameter as 0.1 in all the following the experiments.

For both *coefficient* and *projection* graphs, we set the neighborhood size for similar and different classes as $k_1 = min\{n_i - 1, 15\}$ and $k_2 = n_i - 1$, where n_i is the number of training samples in class i. Figure 5.5b shows the classification results varying the neighborhood size k_1, k_2 on the Extended YaleB face dataset. In this experiment, images are corrupted by 30% occlusion, and we randomly select $n_i = 20$ images per class. As the number of neighbors increases, JP-LRDL achieves better results and using relatively few neighborhoods, remarkably degrades the classification accuracy. There are also two parameters in classification phase as ξ, ω, that we search for their best values in a small set $\{0.001, 0.01, 0.1\}$. Finally, we should note that the lower dimension d is determined during the experiments.

5.6.2 Face Recognition

Extended YaleB Dataset: This dataset [54] contains $2,414$ frontal face images of 38 human subjects captured under different illumination conditions.



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

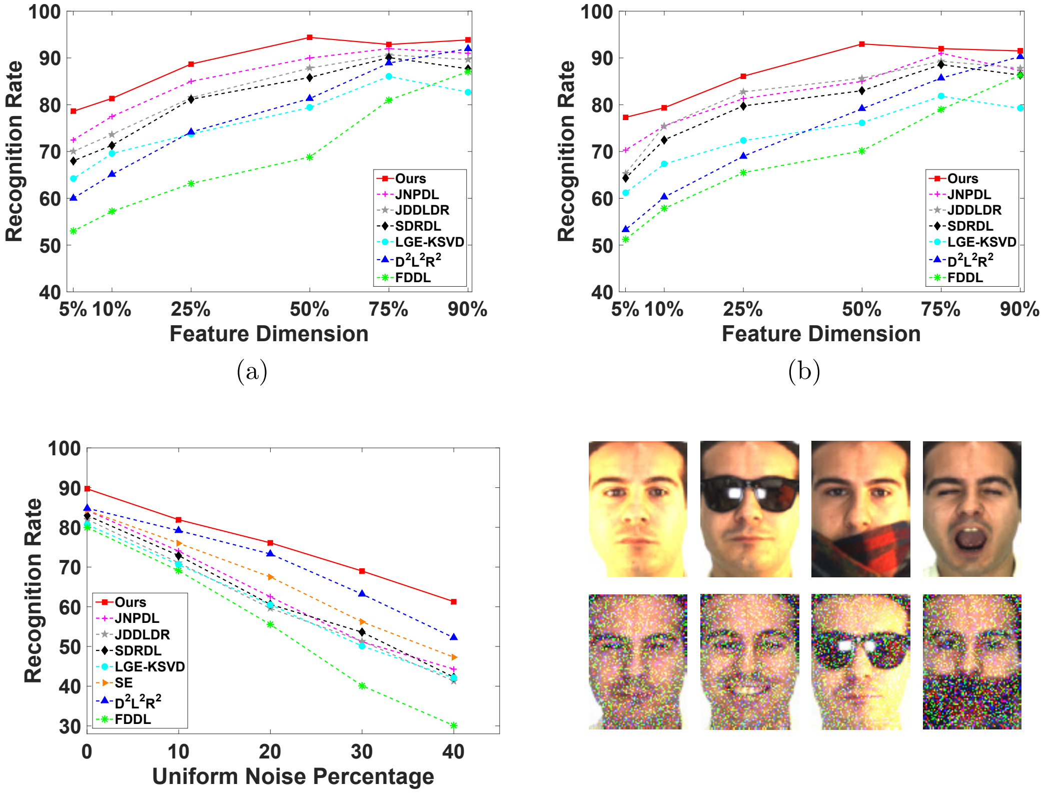
Figure 5.6: Recognition rates (%) of various methods on the Extended YaleB dataset (a) with different levels of pixel corruption (b) with different levels of block corruption (c) with different number of training samples and (d) Samples of the Extended YaleB dataset including original, pixel corrupted (20% and 40%) and occluded (20% and 40%) images

All the face images are cropped and resized to 55×48 and we randomly select 20 images per class for training and the rest is used for test. To challenge our method, we also simulate various levels of corruption and occlusion. For pixel corruption, we replace a certain percentage (from 10% to 50%) of randomly selected pixels of each image with pixel value 255. For occlusion (block corruption), the images are manually corrupted by an unrelated block image at a random location and the percentage of corrupted area is increased from 10% to 50%. Some of the original and corrupted/occluded images of this dataset can be seen in Figure 5.6d. In the following experiments, all the methods utilize the raw images as the feature descriptor, except FDDL and D^{2}L^{2}R^2 methods that initially use PCA to reduce the dimension of features, *i.e.*, the Eigenface is used as input.

We evaluate the robustness of our method to different levels of pixel and block corruption (from 10% to 50%). For each level of corruption, the projected dimension varies between 5% to 90% of the original dimension (2640) and the best achieved result among all dimensions is reported. Figures 5.6a and 5.6b demonstrate that our method consistently obtains better performance than others in all levels of corruption. As the percentage of simulated corruption/occlusion increases, the performance difference between JP-LRDL and other methods becomes more significant. By taking advantage of low-rank and incoherent sub-dictionaries, our method is robust to corruption and illumination variations. These figures also reflect that none of the existing joint dimensionality reduction and dictionary learning methods can achieve good performance for contaminated observations. Equally important, the best performance of JP-LRDL is achieved at 25% of the original dimension, while that of existing joint dimensionality reduction and dictionary learning methods and dictionary learning methods occurs at 50%, and 90% of the original dimensions, respectively. JP-LRDL is superior to other methods when there is a large amount of illumination variations and pixel and/or block corruption, even with fewer features.

We then randomly choose $4 \sim 25$ training samples per subject and evaluate the recognition rate on this dataset. Figure 5.6c shows that our results consistently outperform other counterparts, and significant improvement is observed when there are only a few samples per subject. The proposed JP-LRDL is particularity less sensitive to small-sized dataset and maintains a relatively stable performance even with a few number of training samples.

AR Dataset: The AR face dataset [99] includes over $4,000$ frontal face images from 126 individuals. We select a subset of $2,600$ images from 50 male and 50 female subjects in the experiments. These images include different facial expressions, illumination conditions and disguises. In each session, each person has 13 images, of which 3 are obscured by scarves, 3 by sunglasses and



(a) (b)

(c) (d)

Figure 5.7: Recognition rates (%) of various methods on the AR dataset (a) versus varying feature dimension on Sunglasses scenario (b) versus varying feature dimension on Scarf scenario (c) with different levels of uniform noise on Mixed (Sunglasses+Scarf) scenario and (d) Samples of the AR dataset including original and 20% pixel corrupted images

the remaining ones are of different facial expressions or illumination variations which we refer to as unobscured images. Each face image is resized to 55×40 and following the protocol in [161], experiments are conducted under three different scenarios:

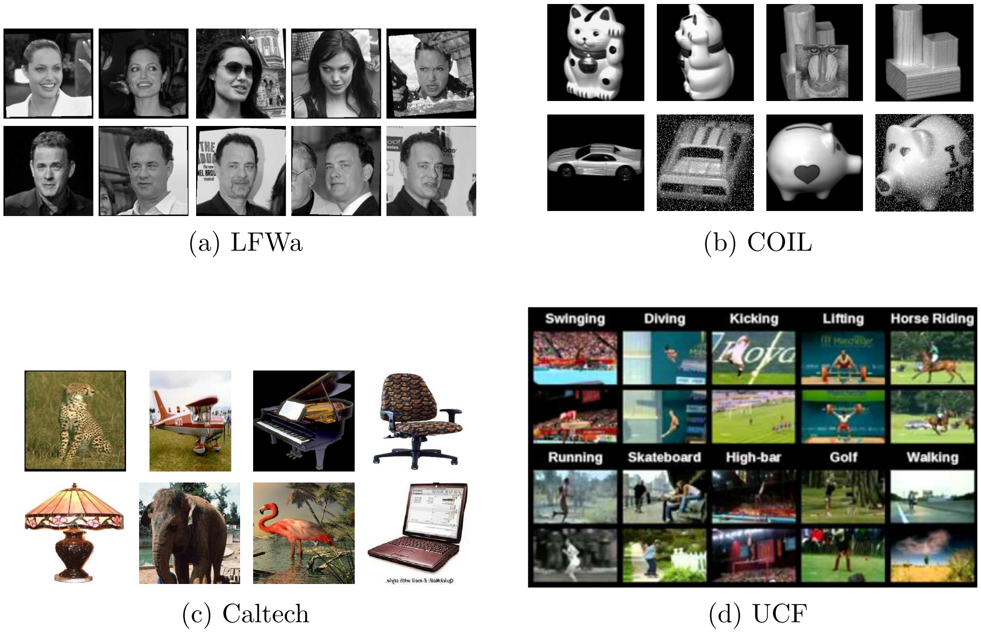
–**Sunglasses:** We select 7 unobscured images and 1 image with sunglasses from the first session as training samples for each person. The rest of unobscured images from the second session and the rest of images with sunglasses are used for testing. Sunglasses occlude about 20% of images.

–**Scarf:** We choose 8 training images (7 unobscured and 1 with scarf) from the first session for training, and 12 test images including 7 unobscured images from the second session, and the remaining 5 images with scarf from two sessions for testing. The scarf covers around 40% images.

−**Mixed (Sunglasses+Scarf):** We consider the case in which both training and test images are occluded by sunglasses and scarf. We select 7 unobscured, plus 2 occluded images (1 with sunglasses, 1 by scarf) from the first session for training, and the remaining 17 images in two sessions for testing per class.

In the following experiments, we use the raw images as the feature descriptor for all the methods, except FDDL and $D^2L^2R^2$, which use Randomface [68], that is generated by projecting a face image onto a random vector. First, we evaluate the robustness of our method in small-sized, large intra-class variability datasets. We consider "Sunglasses" and "Scarf" scenarios and to have more challenge, all the training images are manually corrupted by 20% pixel corruption. Then, we vary the feature dimension from 5% to 90% of the original dimension (2200) and report recognition rates. Figure 5.7d shows some of these original and pixel corrupted images. Figures 5.7a, 5.7b show the recognition rates of JP-LRDL and competing methods over these two scenarios. Our approach achieves the best results compared to the competing methods, across all dimensions and maintains a relatively stable performance in lower dimensions. JP-LRDL is able to achieve the best recognition rate while using 50% of all features. We also note that existing joint dimensionality reduction and dictionary learning methods perform better than dictionary learning methods in lower dimensions, due to the learned projection matrix, which is reasonably more powerful than random projection.

Next, we evaluate our algorithm on the "Mixed" scenario, and to challenge our method, we also simulate uniform noise, such that a percentage of randomly chosen pixels of each image, are replaced with samples from a uniform distribution over $[0; V_{max}]$, where V_{max} is the largest possible pixel value in the image. In this experiment, the projected dimension is fixed as 30% of the original dimension and the recognition accuracy under different levels of corruption is reported in Figure 5.7c. One may infer JP-LRDL shows high robustness to occlusions, severe corruption, illumination and expression changes; however,



(a) LFWa

(b) COIL

(c) Caltech

(d) UCF

Figure 5.8: Sample images of (a) the LFWa dataset (b) the COIL dataset including original, 10% pixel corrupted and 30% occluded images with an unrelated block (c) the Caltech-101 dataset and (d) the UCF dataset

the existing methods fail to handle these variations. Furthermore, the projection graph constraint guarantees the discrimination of projected samples, even in relatively low dimensions.

LFW Dataset: Besides tests with laboratory face datasets, we also evaluate the JP-LRDL on the LFW dataset [64] for unconstrained face verification. LFW contains $13,233$ face images of $5,749$ different individuals, collected from the web with large variations in pose, expression, illumination, clothing, hairstyles, occlusion, etc. Here, we use LWFa dataset [138], which is an aligned version of LFW. We use 143 subject with no less than 11 samples per subject in LFWa dataset ($4,174$ images in total) to perform the experiment. The first 10 samples are selected as the training samples and the rest is for testing. Face images are cropped and normalized to the size of 60×54 and the projected dimension is set as 1000. Also, PCA is used for dimensionality reduction of FDDL and $\text{D}^2\text{L}^2\text{R}^2$ methods. Table 5.1 lists the recognition rates of all the methods, and similar to previous results, JP-LRDL achieves the best performance. These results confirm that the proposed method not only effectively

Table 5.1: Recognition rates (%) of different methods on the LFWa dataset

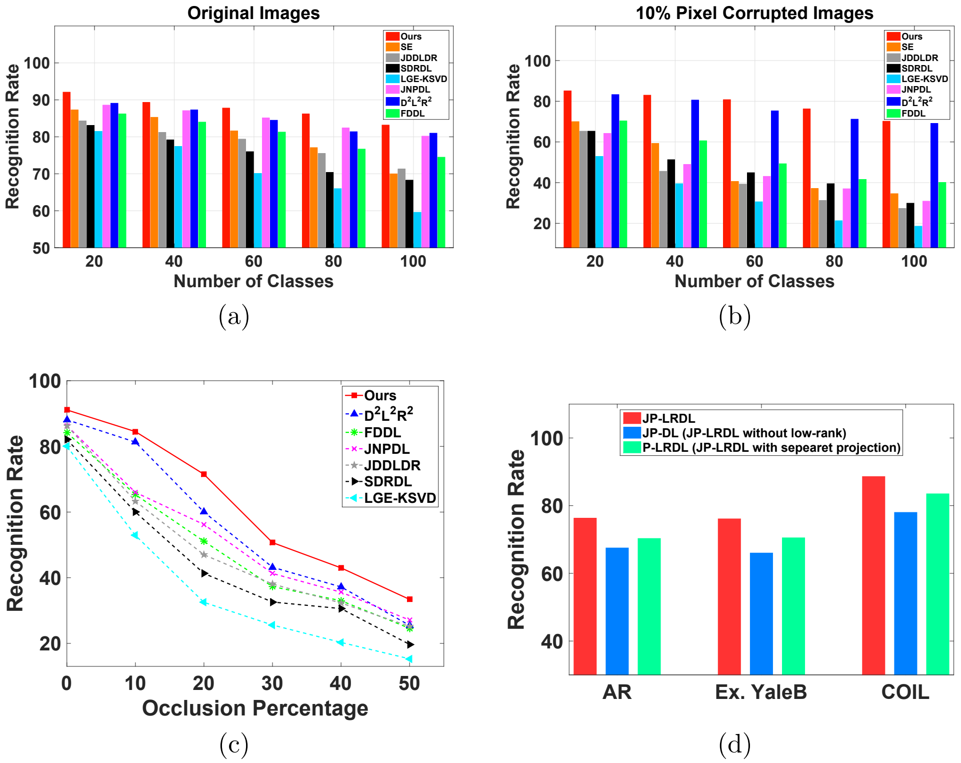
Method	Recognition Rate	Method	Recognition Rate
JNPDL [91]	78.10	JDDLDR [44]	72.40
SDRDL [147]	71.25	LGE-KSVD [111]	70.42
$D^2L^2R^2$ [82]	75.20	FDDL [150]	74.81
SE [103]	76.34	**JP-LRDL**	**79.87**

learn robust feature representations in controlled scenarios, but also have excellent discrimination ability for face images that collected in uncontrolled conditions and have high variation.

5.6.3 Object Recognition

COIL Dataset: The COIL dataset [101] contains various views of 100 objects with different lighting conditions and scales. In our experiments, the images are resized to 32×32 and the training set is constructed by randomly selecting 10 images per object from available 72 images. Some of the original and corrupted images can be found in Figure 5.8b.

We evaluate the scalability of our method and the competing methods by increasing the number of objects (*i.e.*, classes) from 10 to 100. In addition to alternative viewpoints, we also test the robustness of different methods to simulated noise by adding 10% pixel corruption to the original images. Figures 5.9a and 5.9b show the average recognition rates for all compared methods over original images and 10% pixel corrupted images for different class numbers, respectively. Like before, for all the methods, the projected dimension is varied from 5% to 90% of the original dimension (1024), and the best achieved performance is reported. It can be observed that the proposed JP-LRDL outperforms the competing methods and the difference becomes more meaningful, when data are contaminated with simulated noise. All the methods, except $D^2L^2R^2$ and our approach, which utilize low-rank constraint, have difficulty obtaining reasonable results for corrupted data. In particular, our method achieves remarkable performance and demonstrates good scalability in both



(a)

(b)

(c)

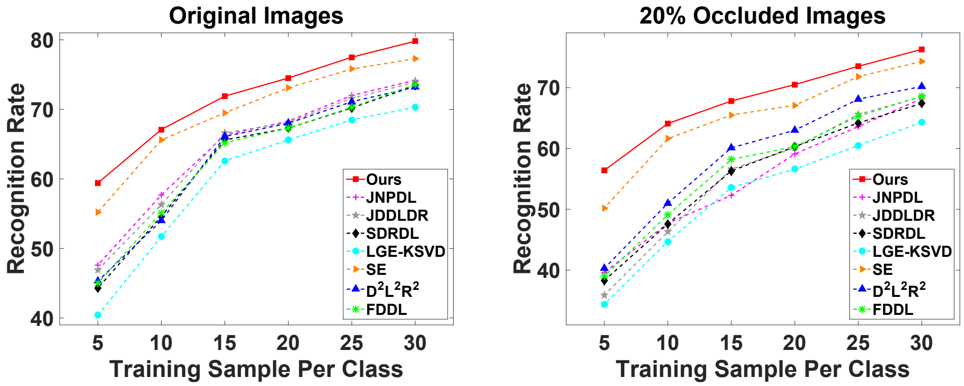
(d)

Figure 5.9: Recognition rates (%) of various methods on the COIL dataset with (a) original images, (b) 10% pixel corrupted images, and (c) versus different levels of occlusion on the COIL-20 dataset (d) The role of different components of JP-LRDL on several datasets

scenarios.

Moreover, we simulate various levels of contiguous occlusion (from 10% to 50%), by replacing a randomly located square block of each test image of the COIL-20 dataset (20 first classes of the COIL dataset), with an unrelated image. We set the feature dimension as 30% of the original dimension, and report the average recognition rates in Figure 5.9c. JP-LRDL achieves the highest recognition rate under different levels of occlusion.

Finally, we design an experiment to show the efficiency of different components of the proposed JP-LRDL framework. To verify the efficacy of low-rank constraint in the framework, we remove $\lambda_3 \sum_{i=1}^{K} \|D_i\|_*$ from Equation (5.11). In a similar fashion, to evaluate the importance of joint dimensionality reduction and dictionary learning process, we remove the projection learning part from JP-LRDL, which means that the projection matrix and structured dictio-



(a) (b)

Figure 5.10: Recognition rates (%) of various methods on the Caltech-101 dataset with different number of training samples on the (a) original images (b) 20% occluded images

nary are learned from training samples separately. We call these two strategies JP-DL and P-LRDL, respectively and compare them with the proposed JP-LRDL on three datasets in Figure 5.9d. In this experiment, the projected dimension is set to 10% of the original dimension and the images are corrupted by 20% block occlusion. For the AR and Extended YaleB datasets, we follow the Mixed scenario and regular experiment protocols, respectively. For the COIL dataset, we utilize first 20 classes. According to the results, once the low-rank regularization is removed, the recognition rate drops significantly in all datasets. Also, we note that JP-LRDL outperforms P-LRDL (with separate projection), and this is mainly due to the fact that some useful information for dictionary learning maybe lost in the separate projection learning phase. The proposed joint learning framework enhances the classification performance, especially when data are highly contaminated and dimension is relatively low.

Caltech-101 Dataset: The Caltech-101 database [43] contains over $9,000$ images from 101 different object categories such as animals, flowers, trees, etc., and 1 background class. The number of images in each class is highly unbalanced, varying from 31 to 800. Figure 5.8c shows some sample images from this dataset. We evaluate our method using dense SIFT-based spatial

Table 5.2: Recognition rates (%) of various methods on the Caltech-101 dataset

Number of Training Samples	15	30
JNPDL [91]	66.83	74.61
JDDLDR [44]	67.70	73.90
SDRDL [147]	65.62	73.25
LGE-KSVD [111]	62.23	70.42
SE [103]	69.50	77.34
$D^2L^2R^2$ [82]	66.10	73.20
FDDL [150]	65.22	73.64
JP-LRDL **without** structural incoherence	66.02	73.71
JP-LRDL **with** structural incoherence	71.97	79.87

pyramid features [68] and set the projected dimension as $3,000$. We run the experiments with 15 and 30 randomly chosen training images per category, and this process is repeated 10 times with different random spits of the training and testing images to obtain reliable results. The final recognition rates are reported as the average of each run in Table 5.2.

In this experiment, to demonstrate the effect of structural incoherence term, we evaluate the recognition rate of JP-LRDL with and without this term. According to the results, our method with structural incoherence term, is superior to other approaches. Incorporating the structural incoherence term, would noticeably enhance the recognition rate, especially in datasets like the Caltech, that has large intra-class variations. Similarly, Figure 5.2 already verified the role of structural incoherence term by presenting the representation error, with and without this term on a subset of the Caltech-101 dataset. The combination of low-rank and incoherence constraints helps us to obtain a better estimate of the underlying distribution of samples, and learn a robust and discriminative subspace. As a result, JP-LRDL is able to recognize objects in images despite imaging variations such as scale, viewpoint, lighting and background.

To verify the robustness of our method to small-sized datasets, we select different numbers of training sample and train it on $\{5, 10, 15, 20, 25, 30\}$ images per category, and test on the rest. To compensate for the variation of the class size, we normalize the recognition results by the number of test images to get per-class accuracies. The final recognition accuracy is then obtained by averaging per-class accuracies across 102 categories. We also repeat this experiment, by replacing a randomly located block of each test image with an unrelated image, such that 20% pixels of every test image is occluded. The recognition rates are reported in Figures 5.10a and 5.10b for the original and occluded images, respectively. Thanks to the efficiency of the proposed JP-LRDL, we achieve superior recognition rate even when the number of training samples is relatively low. Although, the existing methods fail in occluded scenario, our method still maintains satisfactory performance.

5.6.4 Action Recognition

Eventually, we conduct action recognition on the UCF sport action dataset [114]. The video clips in the UCF sport action dataset were collected from various broadcast sports channels. There are 140 videos in total and cover ten sport action classes such as driving, golfing, kicking, lifting, horse riding, where some of which are shown in Figure 5.8d. As the experiment settings in [150], we use action bank features [117] and the projected dimension is reduced to the small number of 100, and this 100-dimensional vector is adopted to represent each video. Like before, FDDL and $D^2L^2R^2$ use PCA for dimensionality reduction. The recognition rates are listed in Table 5.3. Our approach performs the best amongst all the compared methods. In addition, by using the leave-one-video-out experiment setting in [68], the recognition accuracy of JP-LRDL is increased to 98.1%.

5.6.5 Comparison to Deep Learning

Learning through deep neural networks has recently drawn significant attention especially for image classification, and one key ingredient for this success is the use of convolutional architectures. Many variations of CNNs have been

Table 5.3: Recognition rates (%) of different methods on the UCF dataset

Method	Recognition Rate	Method	Recognition Rate
JNPDL [91]	93.03	SE [103]	93.74
JDDLDR [44]	92.54	$D^2L^2R^2$ [82]	92.20
SDRDL [147]	90.62	FDDL [150]	93.60
LGE-KSVD [111]	90.61	**JP-LRDL**	97.52

proposed and demonstrated superior performance over existing shallow methods, in several challenging vision tasks. However, as we will see, such a architecture does not generalize so well to recognition tasks where target dataset is small-sized and diverse in content compared to the base dataset, and also has large intra-class variability.

Since we could not find any work that successfully applies CNN to the same recognition tasks, we use Caffe framework [67] and select a pre-trained network on large-scale ImageNet dataset [34] and then fine-tune it using the target data set for 1000 epochs. We select two popular architectures: AlexNet [75] and VGGNet-D [125]. We evaluate these networks on five target datasets, each of which are known for different kinds of intra-class variation, including illumination and viewpoint changes, occlusion, disguise, background, etc. To challenge these architectures, we also simulate various levels of corruption and occlusion in the target datasets. For all the datasets, we follow the same experiment protocol (*e.g.*, number of train and test samples), as already been described. The evaluation results are given in Table 5.4. Here, n_i shows the number of training samples per class to construct the target dataset.

We observe that these deep architectures do not perform well for none of the face-related experiments, and this becomes worse when simulated corruption or occlusion is present. The reasons could be as follows: The target dataset is smaller in size, but very different in content compared to the original dataset. Recent research [154] reveal that complex models like CNNs are prone to overfitting when the target data is relatively small, and also the effectiveness of feature transfer is declined when the base and target tasks become less similar. In the ImageNet, any kind of human face with very large intra-class variation is categorized as “person and individual” class; however, the target task is face recognition, in which, unique individual should be classified as one class, in spite of pose, expression and illumination changes and occlusion. We notice that, for the object recognition task such as the Caltech-101 dataset, which has more training samples and more similarity with the ImageNet than face datasets, CNNs outperform JP-LRDL; however, when the data are corrupted by simulated noise, their performance drop significantly. One may say, CNNs are not the best model for classification of small-sized datasets with large intra-class variation, especially when the base and target datasets are different in content. It has not escaped our notice that, there could be some deep networks that may have great classification performance for these datasets; but finding the best architecture for these datasets is not a trivial task, and also learning critically depends on expertise of parameter tuning, learning rate selection and so on.

5.7 Summary

In this chapter, we proposed an object classification method for small-sized datasets, which have large intra-class variations. The proposed method simultaneously learns a robust projection and a discriminative dictionary in the low-dimensional space, by incorporating low-rank, structural incoherence and dual graph constraints. These constraints enable us to handle different types of intra-class variability, arising from different lightings, viewpoint and pose changes, occlusion and corruptions, even when only a few training samples per class are available. In the proposed joint dimensionality reduction and dictionary learning framework, learning is performed in the reduced dimensions with lower computational complexity. Besides, by promoting the discriminative ability of the learned projection and dictionary, the projected samples can better preserve the discriminative information in relatively low dimensions; hence, JP-LRDL has superior performance even with a few number of features. Experimental results on different benchmark datasets validated su-

Table 5.4: Recognition rates (%) of deep networks on different scenarios of various datasets

Dataset	Extra Challenge	n_i	AlexNet	VGGNet	JP-LRDL
Ext. YaleB	——	20	43.20	60.54	94.61
Ext. YaleB	20% corruption	20	27.41	41.31	88.01
Ext. YaleB	60% corruption	20	16.40	23.65	54.31
Ext. YaleB	20% occlusion	20	25.43	40.54	89.30
Ext. YaleB	60% occlusion	20	14.51	22.54	64.42
AR-Sunglasses	——	8	30.33	45.10	95.31
AR-Sunglasses	20% corruption	8	15.53	30.24	92.85
AR-Scarf	——	8	30.12	43.90	94.69
AR-Scarf	20% corruption	8	13.04	27.02	92.00
AR-Mixed	——	9	30.17	43.33	95.12
AR-Mixed	20% corruption	9	14.55	28.10	93.00
LFWa	——	10	40.31	57.22	79.87
COIL-20	——	10	65.70	70.76	92.10
COIL-20	30% corruption	10	19.35	36.45	56.72
COIL-20	30% occlusion	10	17.42	34.98	54.33
Caltech-101	——	30	81.15	89.10	79.87
Caltech-101	20% occlusion	30	65.20	69.12	71.00

perior performances of JP-LRDL on image classification tasks especially when those few training samples have large variations.

Chapter 6

Conclusions and Future Directions

6.1 Conclusions

In this thesis, we presented a comprehensive study of sparse learning methods, with a special interest in improving the accuracy of sparse representations of images. According to the sparse representation theory, the linear combination of training samples has great reconstruction power to represent unseen test data. Regarding this fact, different dictionary learning methods have been proposed to train a reconstructive and discriminative dictionary. We studied how dictionaries can be learned in discriminative and representative way for a specific task, and how prior knowledge can be incorporated into the learning framework. Additionally, we addressed several important issues of sparse representation and dictionary learning for pattern recognition, especially image classification by using tools from machine learning, convex optimization and computer vision. In this thesis, we proposed several algorithms based on sparse representations of images, and explored their capability in addressing some challenging tasks in computer vision, which are summarized as follows.

First, we utilized the sparsity concept to solve crowd counting problem. Counting the number of pedestrians in a scene could be difficult in real-world because of severe occlusions and diverse crowd distributions. Considering these challenges, we proposed two robust crowd counting methods and achieved superb performance, especially in large-scale pedestrian datasets. The first method utilizes the image retrieval framework and global image descriptors to perform counting. We also introduced a compact global image descriptor based on compressed sensing theory, to estimate the crowd density. The experimental results reveal that proposed RCS-Count method performs well in estimating the crowd count, but the advantage is particularly significant when the pedestrian dataset is large.

Then, we proposed a more accurate and robust counting method namely SPP-Count, based on the integration of sparse representation-based classification, random projection and semi-supervised elastic net. The proposed SRP-Count method shows superior performance both in small and large pedestrian datasets; however, the advantage is much more remarkable in the latter. With extensive experiments, we demonstrated that if the sparsity is harnessed properly, and the dictionary is rich and large enough to span the testing set variations, then the choice of image descriptors is no longer critical in this framework. Specifically, we showed that the learnt features from a CNN pretrained on object recognition task, can be reused for the purpose of crowd counting. It is commonly believed that SRC requires a rich, diverse and large set of training images to achieve great performance. To fulfill this requirement, we adopt a semi-supervised elastic-net and utilize the sequential information amongst frames, to estimate the count for a large amount of unlabeled images, with only a handful of user-labeled image frames. Experiments on crowd analysis benchmark datasets demonstrated the effectiveness and reliability of the proposed SRP-Count method.

Second, we proposed a supervised feature selection method based on sparsity constraint derived from the decision rule of discriminative dictionary learning. When data are contaminated with severe noise such as occlusion, illumination and pose variations and corruption, the performance of most existing feature selection methods would be limited. So, to identify the most relevant feature subset from the noisy and high-dimensional data, we propose a joint feature selection method using low-rank dictionary learning , called JFS-LDL. We leverage the combination of low-rank matrix recovery and Fisher discrimination dictionary learning to learn discriminative, yet robust dictionary and sparse representations form noisy data. Then, based on the learnt dictionary and coding coefficients, we select features that well preserve the sparse reconstructive relationship of the data and discriminative information, simultaneously. The importance of a feature subset is evaluated by the ratio of intra-class to inter-class reconstruction residual, in the selected subset. By incorporating $l_{2,1}$-norm minimization into the selection objective function, we are able to consider the correlation and interaction amongst features, and select the most discriminative features from the whole feature space, all at once. The proposed JFS-LDL is able to select discriminative features, even from noisy observations. Extensive experiments on benchmark datasets verified the great performance of JFS-LDL for both feature selection and classification tasks. Besides, we adopted the proposed JFS-LDL to count a specific population of tumor cells in microscopic images, and could significantly reduce the estimation error compared to the state-of-the-art methods in the provided dataset.

Finally, we presented a joint projection and low-rank dictionary learning model for object classification in small-sized datasets, which have large intra-class variation. For an effective classification of potentially high-dimensional data, the proposed JP-LRDL simultaneously learns a robust projection matrix and a discriminative dictionary in the low-dimensional space. Based on Fisher discrimination criterion, a structured dictionary whose dictionary atoms have correspondence to the class labels is learned. A graph constraint is imposed on the coding vectors to further enhance class discrimination through making the coding coefficients within the same class to be similar, and the coefficients among different classes to be dissimilar. JP-LRDL introduces low-rank and incoherence promoting constraints on sub-dictionaries to make them more compact and robust to variations, and encourage them to be as incoherent as possible, respectively. Simultaneously, another graph is built on training data to explore intrinsic geometric structure of data, which enables us to preserve the desirable relationship and penalize the unfavorable relationships among training samples, at the same time.

These constraints allow us to handle different types of intra-class variability, arising from different lightings, viewpoint and pose changes, occlusion and corruptions, even when there are a few training samples per class. In the proposed method, learning is performed in the reduced dimensions with lower computational complexity. Besides, by promoting the discriminative ability of the learned projection and dictionary, the projected samples can better preserve the discriminative information in relatively low dimensions; hence, JP-LRDL has superior performance even with a few features. Experimental results on different benchmark datasets validated the superior performance of JP-LRDL on various classification tasks.

6.2 Future Directions

There are several potential future research directions that can be explored to build upon the contributions of this thesis. We describe some of them below.

- First, the proposed SRP-Count method discussed in Chapter 3, can be potentially improved in several ways. Although SRC scheme shows interesting results in different applications including crowd counting, it suffers from major drawbacks such as high computational complexity and low discrimination, due to using all the training samples without any learning. As offered in Chapter 4, these issues can be addressed by learning a smaller-sized and discriminative dictionary through imposing appropriate constraints on the dictionary and coding coefficients. By adopting a discriminative dictionary learning method, we could further decrease the crowd estimation error, even with fewer training samples. This becomes more interesting if we integrate the SSEN objective function into the dictionary learning framework, and rather than learning from only labeled data, exploit the abundant unlabeled data as well. So, we could design an end-to-end efficient semi-supervised dictionary learning method for crowd counting.

  In addition, we may explore the idea of transfer learning in dictionary learning framework. The assumption for this idea is that there is transferrable knowledge in other scenes, which can be employed to further alleviate the burden for data annotation. Although different scenes can be visually very different, the crowd patterns share some common grounds. So, instead of learning the dictionary and sparse representations from scratch in every new scene, the labeled data from other scenes can be utilized to compensate for the lack of labeled data in the new scene, which also facilitates learning of the target model. It should be noted that, in the context of transfer crowd counting we should consider perspective normalization and feature-level alignment issues [23], due to difference of scenes.

- The proposed feature selection method in Chapter 4, currently works in sequential manner *i.e.*, we first exploit low-rank matrix recovery, then FDDL model is applied on the low-rank representation of images, and residual scatter matrices are obtained using the learnt dictionary and sparse coefficients. Finally, the projection matrix is found based on the ratio of residual matrices. Instead of these separated procedures, we can benefit from joint learning similar to Chapter 5, and learn more discriminative features in a more efficient way. The future work may include exploring the joint learning of the feature projection matrix, the dictionary and coding coefficients in a unified framework.

  This idea can be further improved by exploring one more space *i.e.*, sample reduction. Dense features and large-scale data have always been a computational bottleneck of real-life applications. Encouraged by [159], in the future we may introduce a joint learning method that simultaneously learns compact features and removes redundant samples, simultaneously.

- Most of the existing l_1-minimization algorithms are prohibitively costly for real-time computation. This, motivated significant effort in the community to propose non-linear regressors capable of producing good approximations of true sparse codes in a fixed amount of time. In these methods, the basic idea is to design a non-linear, parameterized and feed-forward architecture with a fixed depth, that each of the layers implements a single iteration of a l_1-minimization algorithm such as FISTA [10]. It has been noticed that sparse approximation and deep learning bear certain connections [57]. By turning sparse coding models into deep networks, one may expect faster inference, larger learning capacity, and better scalability. The network formulation also facilitates the integration of task-driven optimization [136].

Moreover, according to our empirical results, the joint projection and low-rank dictionary learning is found superior to other sparse coding methods for classification. Regarding this fact, and rooted in solid literature in trainable networks for approximating sparse codes, the future work may include formulating discriminative low-rank dictionary learning model as a feed-forward neural network, through introducing iterative optimization functions. Since the main objective function includes different terms and variables, one idea is to split it into several iterative algorithms, each capable of approximating one variable. Then, we can design a trainable encoder for each of these variables, whose structure correspond to a few steps of the corresponding optimization objective function.

- It is well established that information fusion using multiple sources can generally improve the recognition performance, since it provides a framework to combine local information from different perspectives, which is more tolerant to the errors of individual sources. Recent years have witnessed a growing interest in multi-modal (multi-view) feature learning techniques; however, very few works have explored it in dictionary learning area. Although some multi-modal dictionary learning methods have been presented, there still exists much room for improvement. The majority of the existing dictionary learning algorithms, including low-rank dictionary learning, are only applicable to single source of data. In the future work, we would like to propose a multi-modal low-rank dictionary learning algorithm for face recognition using heterogeneous sources of information.

  In each modality a discriminative and reconstructive dictionary, and a structured low-rank and sparse representation are learned from face images. Label information from training data is incorporated into the multi-modal learning process by adding an ideal-code regularization term to the objective function, which also encourages collaboration between the modalities. The learnt representations from different modalities are then used for classification directly. We initially intend to use two modalities; for the former we exploit the raw pixels, while the latter is formed by illumination invariant-images [121]. Variable lightings and shadows are amongst the most challenging issues for face recognition, and many methods have been proposed to alleviate these effects. Most recently, Shakeri *et al.* [121] presented an illumination invariant representation of image through decomposing the image into illuminance and reflectance components, and achieved great results for outdoor place recognition in various illumination, shadow and weather variations. Inspired by this success, we hope to increase the face recognition rate under severe illumination and disguise conditions, through multi-modal learning. Our preliminary results validate this claim.

Bibliography

[1] Pets 2009. `http://www.cvg.rdg.ac.uk/PETS2009/`, 2009.

[2] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, 2006.

[3] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[4] Alexandre Alahi, Laurent Jacques, Yannick Boursier, and Pierre Vandergheynst. Sparsity-driven people localization algorithm: Evaluation in crowded scenes environments. In *Performance Evaluation of Tracking and Surveillance (PETS-Winter), 2009 Twelfth IEEE International Workshop on*, pages 1–8. IEEE, 2009.

[5] Antonio Albiol, Maria Julia Silla, Alberto Albiol, and Jose Manuel Mossi. Video analysis using corner motion statistics. In *Proceedings of the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 31–38, 2009.

[6] Senjian An, Wanquan Liu, and Svetha Venkatesh. Face recognition using kernel ridge regression. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE, 2007.

[7] Carlos Arteta, Victor Lempitsky, J Alison Noble, and Andrew Zisserman. Learning to detect cells using non-overlapping extremal regions. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 348–356. Springer, 2012.

[8] Richard G Baraniuk. Compressive sensing. *IEEE signal processing magazine*, 24(4), 2007.

[9] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[10] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[11] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the*

seventh ACM SIGKDD international conference on Knowledge discovery and data mining, pages 245–250. ACM, 2001.

[12] Paul S Bradley and Olvi L Mangasarian. Feature selection via concave minimization and support vector machines. In *ICML*, volume 98, pages 82–90, 1998.

[13] Gabriel J Brostow and Roberto Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 594–601. IEEE, 2006.

[14] Sijia Cai, Wangmeng Zuo, Lei Zhang, Xiangchu Feng, and Ping Wang. Support vector guided dictionary learning. In *European Conference on Computer Vision*, pages 624–639. Springer, 2014.

[15] Emmanuel J Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9):589–592, 2008.

[16] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[17] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[18] Alexey Castrodad and Guillermo Sapiro. Sparse modeling of human actions from motion imagery. *International journal of computer vision*, 100(1):1–15, 2012.

[19] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE, 2008.

[20] Antoni B Chan, Mulloy Morrow, and Nuno Vasconcelos. Analysis of crowded scenes using holistic properties. In *Performance Evaluation of Tracking and Surveillance workshop at CVPR*, pages 101–108, 2009.

[21] Antoni B Chan and Nuno Vasconcelos. Counting people with low-level features and bayesian regression. *Image Processing, IEEE Transactions on*, 21(4):2160–2177, 2012.

[22] Tsung-Han Chan, Kui Jia, Shenghua Gao, Jiwen Lu, Zinan Zeng, and Yi Ma. Pcanet: A simple deep learning baseline for image classification? *IEEE Transactions on Image Processing*, 24(12):5017–5032, 2015.

[23] Chen Change Loy, Shaogang Gong, and Tao Xiang. From semi-supervised to transfer counting of crowds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2256–2263, 2013.

[24] Chih-Fan Chen, Chia-Po Wei, and Yu-Chiang Frank Wang. Low-rank matrix recovery with structural incoherence for robust face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2618–2625. IEEE, 2012.

[25] Ke Chen, Chen Change Loy, Shaogang Gong, and Tony Xiang. Feature mining for localised crowd counting. In *BMVC*, volume 1, page 3, 2012.

[26] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[27] Jen-Tzung Chien and Chia-Chen Wu. Discriminant waveletfaces and nearest feature classifiers for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12):1644–1649, 2002.

[28] Saad Choudri, James M Ferryman, and Atta Badii. Robust background model for pixel based people counting using a single uncalibrated camera. In *Performance Evaluation of Tracking and Surveillance (PETS-Winter), 2009 Twelfth IEEE International Workshop on*, pages 1–8. IEEE, 2009.

[29] Fan RK Chung. Spectral graph theory (cbms regional conference series in mathematics, no. 92). 1996.

[30] Donatello Conte, Pasquale Foggia, Gennaro Percannella, Francesco Tufano, and Mario Vento. A method for counting people in crowded scenes. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 225–232. IEEE, 2010.

[31] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[32] Sanjoy Dasgupta. Experiments with random projection. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 143–151. Morgan Kaufmann Publishers Inc., 2000.

[33] Fernando De La Torre and Michael J Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1-3):117–142, 2003.

[34] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[35] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(4):743–761, 2012.

[36] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.

[37] David L Donoho and Yaakov Tsaig. Fast solution of-norm minimization problems when the solution may be sparse. *IEEE Transactions on Information Theory*, 54(11):4789–4812, 2008.

[38] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification.* John Wiley & Sons, 2012.

[39] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[40] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, 2006.

[41] Kjersti Engan, Sven Ole Aase, and J Hakon Husoy. Method of optimal directions for frame design. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 5, pages 2443–2446. IEEE, 1999.

[42] S Fasanella, E Leonardi, C Cantaloni, C Eccher, I Bazzanella, D Aldovini, E Bragantini, L Morelli, LV Cuorvo, A Ferro, et al. Proliferative activity in human breast cancer: Ki-67 automated evaluation and the influence of different ki-67 equivalent antibodies. *Diagnostic pathology*, 6(1):1, 2011.

[43] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[44] Zhizhao Feng, Meng Yang, Lei Zhang, Yan Liu, and David Zhang. Joint discriminative dimensionality reduction and dictionary learning for face recognition. *Pattern Recognition*, 46(8):2134–2143, 2013.

[45] James Ferryman and Anna-Louise Ellis. Performance evaluation of crowd image analysis using the pets2009 dataset. *Pattern Recognition Letters*, 44:3–15, 2014.

[46] Luca Fiaschi, Ullrich Köthe, Rahul Nair, and Fred A Hamprecht. Learning to count with regression forest and structured labels. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 2685–2688. IEEE, 2012.

[47] François Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5(Nov):1531–1555, 2004.

[48] Homa Foroughi, Nilanjan Ray, and Hong Zhang. People counting with image retrieval using compressed sensing. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4354–4358. IEEE, 2014.

[49] Homa Foroughi, Nilanjan Ray, and Hong Zhang. Robust people counting using sparse representation and random projection. *Pattern Recognition*, 48(10):3038–3052, 2015.

[50] Homa Foroughi, Nilanjan Ray, and Hong Zhang. Object classification with joint projection and low-rank dictionary learning. *arXiv preprint arXiv:1612.01594*, 2016.

[51] Homa Foroughi, Moein Shakeri, Nilanjan Ray, and Hong Zhang. Joint feature selection with low-rank dictionary learning. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 97.1–97.13. BMVA Press, 2015.

[52] Keinosuke Fukunaga. Introduction to statistical pattern recognition. 1990. *Ch*, 9:401–405.

[53] Mehrdad J Gangeh and Ali Ghodsi. On the invariance of dictionary learning and sparse representation to projecting data to a discriminative space. *arXiv preprint arXiv:1503.02041*, 2015.

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

[55] David E Golberg. Genetic algorithms in search, optimization, and machine learning. *Addion wesley*, 1989:102, 1989.

[56] Gene Golub, Stephen Nash, and Charles Van Loan. A hessenberg-schur method for the problem ax+ xb= c. *IEEE Transactions on Automatic Control*, 24(6):909–913, 1979.

[57] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 399–406, 2010.

[58] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[59] Jarvis Haupt, Rui Castro, Robert Nowak, Gerald Fudge, and Alex Yeh. Compressive sampling for signal classification. In *2006 Fortieth Asilomar Conference on Signals, Systems and Computers*, pages 1430–1434. IEEE, 2006.

[60] Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian score for feature selection. In *Advances in neural information processing systems*, pages 507–514, 2005.

[61] Xiaofei He, Deng Cai, Shuicheng Yan, and Hong-Jiang Zhang. Neighborhood preserving embedding. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1208–1213. IEEE, 2005.

[62] Xiaofei He and Partha Niyogi. Locality preserving projections. In *Advances in Neural Information Processing Systems*, pages 153–160, 2004.

[63] Chenping Hou, Feiping Nie, Dongyun Yi, and Yi Wu. Feature selection via joint embedding learning and sparse regression. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1324. Citeseer, 2011.

[64] Gary B Huang, Manu Ramesh, Tamara Berg, and Erik 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, Amherst, 2007.

[65] Kaizhu Huang, Irwin King, and Michael R Lyu. Direct zero-norm optimization for feature selection. In *2008 Eighth IEEE International Conference on Data Mining*, pages 845–850. IEEE, 2008.

[66] Jonathan J. Hull. A database for handwritten text recognition research. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(5):550–554, 1994.

[67] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[68] Zhuolin Jiang, Zhe Lin, and Larry S Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1697–1704. IEEE, 2011.

[69] William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26(189-206):1, 1984.

[70] Kenji Kira and Larry A Rendell. A practical approach to feature selection. In *Proceedings of the ninth international workshop on Machine learning*, pages 249–256, 1992.

[71] Kwangmoo Koh, Seung-Jean Kim, and Stephen Boyd. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, 8(Jul):1519–1555, 2007.

[72] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[73] Shu Kong and Donghui Wang. A dictionary learning approach for classification: separating the particularity and the commonality. In *Computer Vision–ECCV 2012*, pages 186–199. Springer, 2012.

[74] Shu Kong and Donghui Wang. Learning exemplar-represented manifolds in latent space for classification. In *Machine Learning and Knowledge Discovery in Databases*, pages 240–255. Springer, 2013.

[75] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[76] Nojun Kwak and Chong-Ho Choi. Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1):143–159, 2002.

[77] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2006.

[78] Kuang-Chih Lee, Jeffrey Ho, and David J Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on pattern analysis and machine intelligence*, 27(5):684–698, 2005.

[79] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 878–885. IEEE, 2005.

[80] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In *Advances in Neural Information Processing Systems*, pages 1324–1332, 2010.

[81] Jingwen Li, Lei Huang, and Changping Liu. Robust people counting in video surveillance: Dataset and system. In *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, pages 54–59. IEEE, 2011.

[82] Liangyue Li, Sheng Li, and Yun Fu. Discriminative dictionary learning with low-rank regularization for face recognition. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–6. IEEE, 2013.

[83] Min Li, Zhaoxiang Zhang, Kaiqi Huang, and Tieniu Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In *Pattern Recognition, ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008.

[84] Stan Z Li and Juwei Lu. Face recognition using the nearest feature line method. *IEEE transactions on neural networks*, 10(2):439–443, 1999.

[85] Yong Li, Jing Liu, Hanqing Lu, and Songde Ma. Learning robust face representation with classwise block-diagonal structure. *Information Forensics and Security, IEEE Transactions on*, 9(12):2051–2062, 2014.

[86] Zhouchen Lin, Minming Chen, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010.

[87] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):171–184, 2013.

[88] Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 663–670, 2010.

[89] Guangcan Liu and Shuicheng Yan. Latent low-rank representation for subspace segmentation and feature extraction. In *2011 International Conference on Computer Vision*, pages 1615–1622. IEEE, 2011.

[90] Jun Liu, Shuiwang Ji, and Jieping Ye. Multi-task feature learning via efficient l2,1-norm minimization. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 339–348. AUAI Press, 2009.

[91] Weiyang Liu, Zhiding Yu, and Meng Yang. Jointly learning non-negative projection and dictionary with discriminative graph constraints for classification. *arXiv preprint arXiv:1511.04601*, 2015.

[92] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. IEEE, 1999.

[93] Chen Change Loy, Ke Chen, Shaogang Gong, and Tao Xiang. Crowd counting and profiling: Methodology and evaluation. In *Modeling, Simulation and Visual Analysis of Crowds*, pages 347–382. Springer, 2013.

[94] Weizhi Lu, Weiyu Li, Kidiyo Kpalma, and Joseph Ronsin. Sparse matrix-based random projection for classification. *arXiv preprint arXiv:1312.3522*, 2013.

[95] Long Ma, Chunheng Wang, Baihua Xiao, and Wen Zhou. Sparse representation for face recognition based on discriminative low-rank dictionary learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2586–2593. IEEE, 2012.

[96] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Discriminative learned dictionaries for local image analysis. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[97] Julien Mairal, Michael Elad, and Guillermo Sapiro. Sparse representation for color image restoration. *IEEE Transactions on image processing*, 17(1):53–69, 2008.

[98] Julien Mairal, Jean Ponce, Guillermo Sapiro, Andrew Zisserman, and Francis R Bach. Supervised dictionary learning. In *Advances in neural information processing systems*, pages 1033–1040, 2009.

[99] A.M. Martinez and R. Benavente. The AR face database. Technical report, 1998.

[100] Patrenahalli M. Narendra and Keinosuke Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, 100(9):917–922, 1977.

[101] S Nayar, S Nene, and Hiroshi Murase. Columbia object image library (coil 100). *Department of Comp. Science, Columbia University, Tech. Rep. CUCS-006-96*, 1996.

[102] Andrew Y Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78. ACM, 2004.

[103] Hien V Nguyen, Vishal M Patel, Nasser M Nasrabadi, and Rama Chellappa. Sparse embedding: A framework for sparsity promoting dimensionality reduction. In *ECCV 2012*, pages 414–427. Springer, 2012.

[104] Feiping Nie, Heng Huang, Xiao Cai, and Chris H Ding. Efficient and robust feature selection via joint l2,1-norms minimization. In *Advances in Neural Information Processing Systems*, pages 1813–1821, 2010.

[105] H Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 2(47):617–644, 1928.

[106] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[107] Beresford N Parlett. *The symmetric eigenvalue problem*, volume 7. SIAM, 1980.

[108] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[109] Michael Patzold, Rubén Heras Evangelio, and Thomas Sikora. Counting people in crowded environments by fusion of shape and motion information. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 157–164. IEEE, 2010.

[110] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[111] Raymond Ptucha and Andreas E Savakis. Lge-ksvd: robust sparse representation classification. *Image Processing, IEEE Transactions on*, 23(4):1737–1750, 2014.

[112] Vincent Rabaud and Serge Belongie. Counting crowded moving objects. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 705–711. IEEE, 2006.

[113] Ignacio Ramirez, Pablo Sprechmann, and Guillermo Sapiro. Classification and clustering via dictionary learning with structured incoherence and shared features. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3501–3508. IEEE, 2010.

[114] M.D. Rodriguez, J. Ahmed, and M. Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[115] Lorenzo Rosasco, Alessandro Verri, Matteo Santoro, Sofia Mosci, and Silvia Villa. Iterative projection methods for structured sparsity regularization. 2009.

[116] Mark Rudelson and Roman Vershynin. On sparse reconstruction from fourier and gaussian measurements. *Communications on Pure and Applied Mathematics*, 61(8):1025–1045, 2008.

[117] Sreemanananth Sadanand and Jason J Corso. Action bank: A high-level representation of activity in video. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1234–1241. IEEE, 2012.

[118] Santi Seguí, Oriol Pujol, and Jordi Vitria. Learning to count with deep object features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 90–96, 2015.

[119] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[120] Moein Shakeri and Hong Zhang. Corola: a sequential solution to moving object detection using low-rank approximation. *Computer Vision and Image Understanding*, 146:27–39, 2016.

[121] Moein Shakeri and Hong Zhang. Illumination invariant representation of natural images for visual place recognition. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 466–472. IEEE, 2016.

[122] Li Shen, Shuhui Wang, Gang Sun, Shuqiang Jiang, and Qingming Huang. Multi-level discriminative dictionary learning towards hierarchical visual categorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 383–390, 2013.

[123] Xianbiao Shu, Fatih Porikli, and Narendra Ahuja. Robust orthonormal subspace learning: Efficient recovery of corrupted low-rank matrices. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3874–3881. IEEE, 2014.

[124] Terence Sim, Simon Baker, and Maan Bsat. The CMU pose, illumination, and expression database. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(12):1615–1618, 2003.

[125] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[126] RA Smith. Matrix equation xa+bx=c. *SIAM Journal on Applied Mathematics*, 16(1):198–201, 1968.

[127] Ben Tan, Junping Zhang, and Liang Wang. Semi-supervised elastic net for pedestrian counting. *Pattern Recognition*, 44(10):2297–2304, 2011.

[128] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[129] Matthew Turk, Alex P Pentland, et al. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[130] Postma EO van der MLJP and J van den HH. Dimensionality reduction: A comparative review. Technical report, Tilburg, Netherlands: Tilburg Centre for Creative Computing, Tilburg University, Technical Report: 2009-005, 2009.

[131] Chuan Wang, Hua Zhang, Liang Yang, Si Liu, and Xiaochun Cao. Deep people counting in extremely dense crowds. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1299–1302. ACM, 2015.

[132] Haoran Wang, Chunfeng Yuan, Weiming Hu, and Changyin Sun. Supervised class-specific dictionary learning for sparse modeling in action recognition. *Pattern Recognition*, 45(11):3902–3911, 2012.

[133] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367. IEEE, 2010.

[134] Li Wang, Ji Zhu, and Hui Zou. Hybrid huberized support vector machines for microarray classification. In *Proceedings of the 24th international conference on Machine learning*, pages 983–990. ACM, 2007.

[135] Z Wang, J Yang, N Nasrabadi, and T Huang. Look into sparse representation-based classification: A margin-based perspective. In *IEEE International Conference on Computer Vision (ICCV), Sydney*, pages 759–769, 2013.

[136] Zhangyang Wang, Qing Ling, and Thomas S Huang. Learning deep l0 encoders. *arXiv preprint arXiv:1509.00153*, 2015.

[137] Jason Weston, André Elisseeff, Bernhard Schölkopf, and Mike Tipping. Use of the zero-norm with linear models and kernel methods. *Journal of machine learning research*, 3(Mar):1439–1461, 2003.

[138] Lior Wolf, Tal Hassner, and Yaniv Taigman. Similarity scores based on background samples. In *Asian Conference on Computer Vision*, pages 88–97. Springer, 2009.

[139] John Wright, Yi Ma, Julien Mairal, Guillermo Sapiro, Thomas S Huang, and Shuicheng Yan. Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE*, 98(6):1031–1044, 2010.

[140] John Wright, Allen Y Yang, Arvind Ganesh, Shankar S Sastry, and Yi Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2009.

[141] Tong Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, pages 224–244, 2008.

[142] Xinyu Wu, Guoyuan Liang, Ka Keung Lee, and Yangsheng Xu. Crowd density estimation using texture analysis and learning. In *2006 IEEE international conference on robotics and biomimetics*, pages 214–219. IEEE, 2006.

[143] Ying Nian Wu, Zhangzhang Si, Haifeng Gong, and Song-Chun Zhu. Learning active basis model for object detection and recognition. *International journal of computer vision*, 90(2):198–235, 2010.

[144] Hui Yan. Sparsity preserving score for joint feature selection. In *Intelligence Science and Big Data Engineering*, pages 635–641. Springer, 2013.

[145] Hui Yan and Jian Yang. Sparse discriminative feature selection. *Pattern Recognition*, 48(5):1827–1835, 2015.

[146] Allen Y Yang, S Shankar Sastry, Arvind Ganesh, and Yi Ma. Fast l1-minimization algorithms and an application in robust face recognition: A review. In *2010 IEEE International Conference on Image Processing*, pages 1849–1852. IEEE, 2010.

[147] Bao-Qing Yang, Chao-Chen Gu, Kai-Jie Wu, Tao Zhang, and Xin-Ping Guan. Simultaneous dimensionality reduction and dictionary learning for sparse representation based classification. *Multimedia Tools and Applications*, pages 1–22, 2016.

[148] Meng Yang. *Regularized robust coding and dictionary learning for face recognition*. PhD thesis, The Hong Kong Polytechnic University, 2012.

[149] Meng Yang, Dengxin Dai, Lilin Shen, and Luc Gool. Latent dictionary learning for sparse representation based classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4138–4145, 2014.

[150] Meng Yang, Lei Zhang, Xiangchu Feng, and David Zhang. Fisher discrimination dictionary learning for sparse representation. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 543–550. IEEE, 2011.

[151] Meng Yang, Lei Zhang, Xiangchu Feng, and David Zhang. Sparse representation based fisher discrimination dictionary learning for image classification. *International Journal of Computer Vision*, 109(3):209–232, 2014.

[152] Meng Yang, Lei Zhang, Jian Yang, and David Zhang. Metaface learning for sparse representation based face recognition. In *2010 IEEE International Conference on Image Processing*, pages 1601–1604. IEEE, 2010.

[153] Yi Yang, Heng Tao Shen, Zhigang Ma, Zi Huang, and Xiaofang Zhou. l2, 1-norm regularized discriminative feature selection for unsupervised learning. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1589. Citeseer, 2011.

[154] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[155] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863, 2003.

[156] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[157] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. Cross-scene crowd counting via deep convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 833–841, 2015.

[158] Haichao Zhang, Yanning Zhang, and Thomas S Huang. Simultaneous discriminative projection and dictionary learning for sparse representation based classification. *Pattern Recognition*, 46(1):346–354, 2013.

[159] Man Zhang, Ran He, Dong Cao, Zhenan Sun, and Tieniu Tan. Simultaneous feature and sample reduction for image-set classification. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[160] Qiang Zhang and Baoxin Li. Discriminative k-svd for dictionary learning in face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2691–2698. IEEE, 2010.

[161] Yangmuzi Zhang, Zhuolin Jiang, and Larry Davis. Learning structured low-rank representations for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 676–683, 2013.

[162] Yin Zhang. Recent advances in alternating direction methods: Practice and theory. In *IPAM Workshop on Continuous Optimization*, 2010.

[163] Tao Zhao, Ram Nevatia, and Bo Wu. Segmentation and tracking of multiple humans in crowded environments. *IEEE transactions on pattern analysis and machine intelligence*, 30(7):1198–1211, 2008.

[164] Zheng Zhao, Lei Wang, Huan Liu, et al. Efficient spectral feature selection with minimum redundancy. In *AAAI*, 2010.

[165] Ning Zhou, Yi Shen, Jinye Peng, and Jianping Fan. Learning interrelated visual dictionary for object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3490–3497. IEEE, 2012.

[166] Ji Zhu, Saharon Rosset, Trevor Hastie, and Rob Tibshirani. 1-norm support vector machines. *Advances in neural information processing systems*, 16(1):49–56, 2004.

[167] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE, 2004.