Compressive Sensing for Computer Vision and Image Processing

by

Naveen Kulkarni

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved May 2011 by the
Graduate Supervisory Committee:

Baoxin Li, Chair
Jieping Ye
Arunabha Sen

ARIZONA STATE UNIVERSITY

August 2011

ABSTRACT

With the introduction of compressed sensing and sparse representation, many image processing and computer vision problems have been looked at in a new way. Recent trends indicate that many challenging computer vision and image processing problems are being solved using compressive sensing and sparse representation algorithms. This thesis assays some applications of compressive sensing and sparse representation with regards to image enhancement, restoration and classification. The first application deals with image Super-Resolution through compressive sensing based sparse representation. A novel framework is developed for understanding and analyzing some of the implications of compressive sensing in reconstruction and recovery of an image through raw-sampled and trained dictionaries. Properties of the projection operator and the dictionary are examined and the corresponding results presented. In the second application a novel technique for representing image classes uniquely in a high-dimensional space for image classification is presented. In this method, design and implementation strategy of the image classification system through unique affine sparse codes is presented, which leads to state of the art results. This further leads to analysis of some of the properties attributed to these unique sparse codes. In addition to obtaining these codes, a strong classifier is designed and implemented to boost the results obtained. Evaluation with publicly available datasets shows that the proposed method outperforms other state of the art results in image classification. The final part of the thesis deals with image denoising with a novel approach towards obtaining high quality denoised image patches using only a single image. A new technique is proposed to obtain highly correlated image patches through sparse representation, which are then subjected to matrix completion to obtain high quality image patches. Experiments suggest that there may exist a structure within a noisy image which can be exploited for denoising through a

low-rank constraint.

For my Parents and Sister.

ACKNOWLEDGEMENTS

I am indebted to several people who have contributed directly or indirectly to this thesis.

Words cannot express my greatest gratitude to my advisor, Professor Baoxin Li for his unreserved support, elaborate guidance, and inspiring thoughts and encouragement throughout my M.S study. I have tried to emulate his approach towards problem solving and how to formulate solve and present research. His enthusiastic outlook to research and his boundless energy continue to inspire me. My interactions with him have positively benefitted me and will hold me in good stead for the rest of my career.

I would like to thank my committee members, Professor Jieping Ye and Professor Arunabha Sen for serving as members of my thesis committee and examining my thesis report.

I would also like to thank my friends and colleagues at Arizona State University and beyond. I would particularly like to thank Adarsh Bhatt, Srikanth M.G, Abhishek Khade, Sudhir Kumar Srinivasan and Pradeep Nagesh who have stood by me through all my good and difficult times.

It is hard to express the gratitude I owe to my family in words.This would not have been possible without their support. I wish to thank my parents and my sister for their endless love and unconditional support.The motivation and encouragement I received from my family has been the biggest factor for completion of my Masters degree.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

List of Symbols

Chapter 1

INTRODUCTION

Imaging and computer vision have been two extensively researched areas which have directly or indirectly contributed to the technological advancement in visual computing. Image representation, recognition, modeling, enhancement, restoration, analysis and reconstruction from projections have been few of the areas which have been looked at in a different way after the introduction of Compressive Sensing. With the plethora of data available, it is very important to choose which datum to pick from the vast set of data. Recently developed compressed sensing provides direction in selecting the most important data. The challenging task of computer vision has been and will be to develop systems which mimic, represent and analyze the behavior charaterized by human beings.The systems which aim at understanding and representing such behavior should have highly accurate sensing and acquisition capabilities. This must be followed by certain pre-processing for input data formatting, actual methodology of feature formation and analysis, followed by post-processing such as enhancement and restoration. The following steps outline some of the steps involved in a typical computer vision system. Although different systems are application dependent, most of them can be generalized to comprise of the following underlying steps.

Image Acquisition:Also commonly known as imaging is the first stage involved in a computer vision system.A computational model of a camera, atleast for its geometric part tells how to project a natural 3D scene onto an image and how to project back from the image to 3D. There are different Camera models classified according to different criteria such as viewpoint, complexity and imaging type. Two plane model, fisheye model, affine models are some of the commonly used camera models in the computer vision systems. A CCD or a CMOS sensor is invariably used in most of the spatially sampled imaging systems with a

pre-defined set of points defined on the imaging plane which follow the Shannon-Nyquist sampling theorem. Sampling of amplitudes also known as quantization and temporal sampling defined by the frame-rate are also involved in the acquisition process.

Pre-processing: Before a computer vision method can be applied to an image in order to extract certain features, it is usually necessary to format the data in such a way as to satisfy certain criterion required by the method. Some of the examples include

- Re-sampling to make sure that the working image co-ordinate system is accurate.
- Image restoration method such as noise reduction to ascertain that sensor noise does not falsify the actual data values.
- Contrast stretching and enhancement to obtain relevant information before any method is acted upon.
- Scaling and normalization for appropriate scale-space representation.

Image Feature extraction: Feature extraction and selection has been an active area of research in computer vision, machine learning, data mining ,text mining, genomic analysis , image retrieval etc. Image features have different complexities depending on the input image type. Stable feature selection, optimal redundancy removal and exploiting auxiliary data are some of the important challenges associated with feature selection. There are various types of features such as spatial features, transform based features, edges and boundaries, shape features, textures etc. Feature extraction is an important step for analysis of image data. It also plays an important role in further post-processing and recognition/classification purposes as well.

Image Segmentation and Recognition/Classification: Image segmentation refers to the decomposition of a scene into its components. It is one of the important steps in image analysis. Various segmentation techniques such as amplitude thresholding, component labeling, boundary based approaches, region based clustering, template matching and texture segmentation are extensively used in image analysis which leads to recognition/classification. Segmentation makes sure that all the irrelevant features are discarded out paving way for selection of useful objects of interest. Classification is the final step which quantifies the nature of data and leads to decision making. As the term itself indicates, it is used to classify the object into one of several categories. Classification and segmentation are closely intertwined with each one aiding the other in the final outcome.At a higher level classification can be either supervised or unsupervised. Supervised classification does not depend on a priori probability distribution functions and are based on reasoning and heuristics. In unsupervised learning,the idea is to indentify the clusters or natural groupings in the feature space. A cluster is a set of points in feature space for which their local density is large compared to the density of feature points in the surrounding region.Clustering techniques are useful for image segmentation and also for classification of raw data to establish different classes.

1.1 Motivation and direction of thesis

One of the main motivation for developing new computer vision applications is the recent introduction of compressive sensing. With the advent of compressive sensing a large number of new methods have been developed for image analysis in computer vision. This particular work derives mathematical formulations from the recently developed compressive sensing, sparse representation and matrix completion for related applications in image processing and computer vision. While image acquistion and pre-processing plays an important role in acquring raw input data, image analysis, image restoration and image enhancement are three impor-

tant aspects of a computer vision rendering system. Image analysis system which consists of feature extraction, segmentation and classification/recognition forms the first important step of understanding the raw image data. The analyzed data is useful in making decisions in general applications such as video surveillance for event and activity detection, organizing information for content based data retrieval, for computer human interaction etc.

Of all the visual tasks we might expect a computer to perform, analyzing a scene and recognizing all of the constituent objects remains the most challenging. While computers excel at accurately reconstructing the 3D shape of a scene from images taken from different views, they cannot name all the objects present in the image.Then the question that arises is, why is recognition so hard? The real world is made of innumerable objects which all occlude one another, have variable poses, exhibit variability in terms of sizes,shapes and appearance. Thus it remains an extremely hard problem of just performing an exhausting matching against a database of exemplars. The most challenging version of recognition is general category object recognition. Some techniques may rely purely on the presence of features(such as bag of words or visual words or SIFT features),while other methods involve segmenting the image into semantically meaningful regions so as to obtain unique regions for classification. Given such an extremely rich and complex nature of the topic,there is a need to divide the problem into subsequent smaller steps before an effort is made to solve each one of them individually and the problem as a whole.

General object recognition falls into two broad categories, namely the instance recognition and class recognition. Instance recognition involves recognizing a known 2D or 3D rigid object,potentially being viewed from a novel viewpoint, against a cluttered background and with partial occlusions [74].The class recognition is a much harder problem of recognizing any instance of a particular object

such as animals,any general surrounding objects etc. The harder problems typically are characterized by a large dataset. Computational complexity is extremely high if all of the data is to be used for recognition/classification. Compressive sensing would play a handy role in such a scenario. Image data is invariably sparse, leading to representations which can be much less denser than the ones involving large raw inputs. Thus sparse representation would be able to convert such dense data into sparse data.

Sparse signal representation has proven to be an extremely powerful tool for acquiring, representing and compressing signals. The success is predominantly due to the fact that general audio, image ,video signals have naturally sparse representations in a basis (such as DCT, wavelets etc) or a concatenation of such bases.This successful technique which has played an extremely important role in classical signal processing for compact representations can also be employed to computer vision applications where contents and semantics of the image are more important than representations and recovery.This thesis tries to capture the essence of compressive sensing based sparse representation which can be successfully employed in generic image processing enhancement technique such as image Super-Resolution(SR) and image restoration such as image denoising and also in computer vision applications such as Image classification.

With this background and motivation, the emphasis of this thesis is on the following topics:

(i) *Super-Resolution*: Redundant representations of randomly sampled dictionaries have provided good performance in sparse representation based reconstruction algorithms. In this thesis, experimentation and analysis of redundant representations based trained dictionaries is conducted. In addition to analysis, it also provides insights into the properties of these dictionaries and its relation to compressive sensing. Also an empirical analysis of results for recovery and representa-

tion based Super-Resolution is provided. In addition to these the sparse solution space for representation and recovery methods is analyzed and zone of operation for a trade-off between sparsity and reconstruction fidelity is provided.

(ii) *Image Classification*: Another computer vision application which is looked into is image classification.Image classification has been an extensively researched area in the last few years.It forms an important part of object recognition.Different models and methods have been analyzed in the past few years,but none of them have been able to achieve high degree of accuracy through these methods. A new approach towards image classification through the method of obtaining trained dictionaries through sparse representation in an affine invariant feature space is depicted.Through the combination of a good classifier and good feature repesentation state of the art results on Caltech-101 and Caltech-256 dataset are presented.

(iii) *Image Denoising*:The last part of the thesis deals with one of the classical image restoration technique namely image denoising. Inexact recovery of a large matrix through matrix completion has provided new insights into the way missing data can be recovered among a large set of correlated data.In this thesis, experimentation and analysis of sparse representation based noise recovery is carried out. In addition to obtaining noisy sparse representations of a noisy image, noisy pixel elimination through matrix completion is analyzed and understood.

1.1.1 Related Work

This section reviews some of the common fundamental principles utilized in super-resolution, image classification and image denoising. An overview of compressive sensing and sparse representation from a compact representation point of view is dealt with in detail. First the representation and recovery methods of image super-resolution and image denoising is discussed. Then the focus shifts towards the compact feature representation in image classification. But before that we shall look into the emergence of compressive sensing commonly abbreviated as

"CS".

1.2 Compressive Sensing(CS): A Background

The Shannon/Nyquist sampling theorem specifies that to avoid losing information when capturing a signal, one must sample at least two times faster than the signal bandwidth. In many application, including digital image and video cameras, the Nyquist rate is so high that too many samples are obtained,making compression a necessity prior to storage or transmission. In other practical applications, including imaging systems and high speed analog to digital converters, increasing the sampling rate is very expensive.This section surveys the theory of compressive sampling also known as compressive sensing or CS, a novel sensing/sampling paradigm.CS theory asserts that one can recover certain signals and images from far fewer samples or measurements that traditional methods use [72]. For this to happen, CS relies on two principles: sparsity, which pertains to the properties of natural signals of interest, and incoherence, which involves how signal is sensed/sampled.

The information rate of a continuous time signal may be much smaller than that suggested by its bandwidth. This is the principle used to express the notion of sparsity.This can also be stated in terms of a discrete-time signal wherein the number of degrees of freedom of the signal is comparably much smaller than its length. General natural signals are sparse or compressible and when expressed in an appropriate basis Ψ have compact representations. This is the principle which CS exploits.

Incoherence extends the duality between time and frequency. It expresses the idea that objects having a sparse representation in Ψ must be spread out in the domain in which they are acquired. This is similar to the analogy in which Dirac or a spike in the time domain, is spread out in the frequency domain. But in order for the signal of interest to be sparse in Ψ,incoherence suggests that the

sampling/sensing waveforms have an extremely dense representation in Ψ.

The important observation is that one can efficiently design good sensing/sampling protocols that captures all the relevant information from natural occuring sparse signals and compress it into a much smaller data. The acquisition signals or the waveforms need not be modifiable and hence need not necessitate adaptive sparsifying basis. Thus with a small amount of fixed waveforms with lot of incoherency with the signal to be acquired, an efficient design strategy can be devised to capture the sparse information. Without trying to understand the signal, these sampling protocols capture the information very efficiently.Numerical optimization provides a mechanism to reconstruct or recover the signal completely from small amount of data collected.Thus using an imcomplete set of measurements, compressive sensing is able to sample the signal at an information rate and power which is much lower than that defined by shannon/nyquist theorem.

Compressive sensing which was originally developed for single pixel camera and for medical imaging and ADC systems has been subsequently adopted into the general signal processing community. Built upon the groundbreaking work by [16] and [29], based on a new set of paradigms on signal model compared to the existing Shannon/Nyquist model. The new paradigms, which CS theory is built upon and are different from the conventional Shannon/Nyquist notion according to the following:

1.Measurement principle

2.Sparsity

3.Incoherence

4.Measurement systems and sparse signal recovery conditions

5.Reconstruction Algorithms in CS Decoders

1.2.1 Measurement principle

Unlike in Shannons sampling case, there is no concept of point samples for representing the signal. However, linear measurements of the signal are obtained which are now a generalization of samples, obtained by projection into a different space called the measurement space. There are no actual pixels involved in an image here since the captured information constitute a linear set of measurements. A property called incoherence is necessary for acquiring good linear measurement in the new measurement space defined in reference to the transformation space (discussed in detail later). Under these two paradigms, the following section provides explanation of CS theory from a mathematical point of view. Here and in most part of this document, only the discrete case of CS (called the discrete CS) is considered. Let f(t) be a signal obtained by linear functionals

$$y_k = \langle f, \phi_k \rangle k = 1, \ldots, m \tag{1.1}$$

With the basis functions ϕ_k we wish to correlate the signal to be acquired for a fixed m. The sensing waveforms can be Dirac delta functions(spikes) or sinusoids. A total of m such correlations using m different sensing waveforms lead to m measurement values which are collaboratively called the new linear measurements. At this point in time we would restrict our attention to discrete signals $f \in \Re^n$. Now we are concerned with undersampled situations in which the number m of available measurements is much smaller than the dimension n of the signal f. This raises an important question about accurate reconstruction from $m << n$ measurements only. This can be achieved through the set of operations given by

$$y = Af, y \in \Re^M, f \in \Re^N, A \in \Re^{MxN} \tag{1.2}$$

Though the problem is ill-posed in general, a way out can be found by relying on realistic models of objects f which naturally exist. In CS terminology, $\mathbf{y} =$

$[y_1, y_2, \ldots, y_M]^T \in \Re^M$ is the measurement vector, $\Phi = [\phi_1, \phi_2, \ldots, \phi_M]^T$ is the new measurement space called the measurement matrix. Suppose f is a compressible signal which is K-sparse(K<N), with the sparse representation expressed in an orthonormal transformation space defined by $\Psi \in \Re^{NxN}$, then f can be expressed through the orthonormal basis $\Psi = [\psi_1, \psi_2, \ldots, \psi_n]$ as follows:

$$f = \sum_{i=1}^{n} x_i \psi_i \tag{1.3}$$

where x is the coefficient sequence of f, $x_i = \langle f, \psi_i \rangle$. With this background we move to the next important concept Sparsity.

1.2.2 Sparsity

The Eq.1.3 described above represents an expansion of the signal in terms of a few coeffcients of a basis function. Now sparsity implies that when a signal has a sparse expansion, one can discard the smaller coefficients without losing out any perceptually meaningful information. Now if we consider f_K obtained by the keeping the K largest values of x_i in the expansion Eq.1.3, then this vector x_K is sparse in a strict sense since all but a few of its entries are zero. Since Ψ is an orthonormal basis , we have $\| f - f_K \|_{l2} = \| x - x_K \|_{l2}$ and if x is sparse or compressible, then x is well approximated by x_K and thus the error $\| f - f_K \|_{l2}$ is small. This principle has been very effective in JPEG-2000 [30] and others since there would not be any perceptual loss of information and also the gains attained in terms of compression efficiency is high. In general sparsity is an efficient modeling tool which permits effective signal processing as in the case of statistical estimation and classification, efficient data compression and so on.Sparsity has significant carriage on the acquisition process itself and it determines efficient acquisition of signals nonadaptively [72].

1.2.3 Incoherence

Suppose we have two pairs of orthobases Ψ ,Φ of $\Re^n$ and Φ is used for sensing f and the other orthobasis Ψ is used for representing f. The coherence between the sensing basis Φ and the representing basis Ψ is given by μ through the following equation:

$$\mu(\Phi, \Psi) = \sqrt{n}. \max_{1 \leq k,j \leq n} |\langle \phi_k, \psi_j \rangle|. \tag{1.4}$$

Coherence measures the correlation between any two basis vectors of the orthonormal bases Φ and Ψ [31]. Now if ϕ and ψ contain correlated elements, the coherence is large else its small and the range of coherence $\mu(\Phi, \Psi) \in [1, \sqrt{n}]$. A compressive sampling based acquisition is mainly concerned with low coherence pairs. For example a dirac delta and a sinusoid are maximally incoherent in any dimension and the pair obeys $\mu(\Phi, \Psi) = 1$. In general random matrices are largely incoherent with any fixed basis Ψ and it follows that higher the incoherence, the lower the number of samples necessary for perfect recovery.

1.2.4 Measurement systems and sparse signal recovery conditions

We would like to measure all the n coefficients of f, but we get to observe only a subset of the samples $\mathrm{M} \subset 1, 2 \ldots, n$. Now these subset of samples are encoded in the vector given by the following:

$$y_k = \langle f, \phi_k \rangle, k \in M \tag{1.5}$$

We now try to recover the signal f through the reconstruction equation $\tilde{f} = \Psi \tilde{x}$ where $\tilde{x}$ is the solution obtained through l_1-norm minimization through the convex optimization program given by

$$\min_{\tilde{x} \in \Re^n} \| \tilde{x} \|_{l1} \; subject \; to \; y_k = \langle \phi_k, \Psi \tilde{x} \rangle, \forall \; k \in M \tag{1.6}$$

Thus among all signals $\tilde{f} = \Psi \tilde{x}$ we pick the appropriate coefficient sequence which has the lowest l_1 norm.Suppose the signal $f \in \Re^n$ in terms of the coefficient x is

K sparse, then selecting m measurements in the Φ domain uniformly at random gives the following:

$$m \geq C.\mu^2(\Phi, \Psi).K.logn \tag{1.7}$$

for some positive constant C, the solution to Eq. 1.6 is exact with overwhelming probability. Also the probability of success exceeds (1-δ) if $m \geq C.\mu^2(\Phi, \Psi).K.$ $\log(n/\delta)$. An immediate inference based on this equation is that the role of coherence is very simple;the smaller the coherence, the fewer samples are needed,and hence we look for systems with low coherence.Also there would be no loss of information by measuring just any set of m coefficients which may be far less than the signal size and moreover if $\mu(\Phi, \Psi)$ is equal or close to one, then for a K-sparse signal, K.log$n$ samples are sufficient instead of n. Also the signal f can be exactly recovered from smaller data set through minimizing a convex functional which need not have any knowledge about number of nonzero coefficients and their locations or values.

Restricted Isometric Property: Another important concept in the study of general principles of CS is the restricted isometric property(RIP) [32]. For each integer K = 1,2,. . . , define the isometry constant δ_K of a matrix A as the smallest number such that

$$(1 - \delta_K) \parallel x \parallel_{l2}^2 \leq \parallel Ax \parallel_{l2}^2 \leq (1 + \delta_K) \parallel x \parallel_{l2}^2 \tag{1.8}$$

holds good for all K-sparse vectors *f*. Without strict implications, it can be said that matrix A obeys the RIP of order K if δ_K is not too close to one.Because of this property, the matrix A preserves the Euclidean length of K-sparse signals, which implies that K-sparse vectors cannot be in the null space of A. A notion of pseudo-orthogonality can be developed from this theory wherein any subset of the K-columns of the matrix A are approximately orthogonal. Now to see a connection between RIP and CS, suppose we acquire K-sparse signals with A and

δ_{2K} is sufficiently less than one.This implies that all pairwise distances between K-sparse signals must be well preserved in the measurement space [72].

$$(1-\delta_{2K}) \parallel x_1 - x_2 \parallel_{l2}^2 \leq \parallel Ax_1 - Af_2 \parallel_{l2}^2 \leq (1+\delta_{2K}) \parallel f_1 - f_2 \parallel_{l2}^2 \tag{1.9}$$

holds true for all K-sparse vectors x_1,x_2.

1.2.5 Reconstruction Algorithms as CS Decoders

The objective of the CS decoder is to reconstruct the K-sparse signal $f \in \Re^N$ from its compressive measurements $y \in \Re^M$. The first method of solving this l_1 optimization problem is through Basis Pursuit(BP).It is given by

$$\tilde{x} = argmin \parallel x \parallel_1 s.t.\ y = \Phi\Psi x \tag{1.10}$$

Yet another method of reconstruction through Basis Pursuit Denosing(BPDN) is well suited in cases where measurements are noisy. The measuring process with noise can be given by

$$y = \Phi x + z,\ \ y \in \Re^M, x \in \Re^N, \Phi \in \Re^{MxN} \tag{1.11}$$

where z is a stochastic noise or a deterministic unknown error term.The solution to the BPDN optimization problem is given by

$$\tilde{x} = argmin \parallel x \parallel_1 s.t. \parallel y - \Phi\Psi x \parallel < \epsilon \tag{1.12}$$

where ϵ is a constant which takes into account the variance of the noise z. An unconstrained version of the above BPDN is given by the following equation given by

$$\tilde{x} = argmin\ \tau \parallel x \parallel_1 + 0.5 \parallel y - \Phi\Psi x \parallel_2^2 \tag{1.13}$$

The reason the unconstrained version is popular is mainly due to its faster solving capability. A faster algorithm for solving the sparse optimization problem is called the Sparse Receovery by Separable Approximation(SpaRSA) [33] which controls the tradeoff between sparsity of coefficients and fidelity of the reconstruction.More details about this is discussed in the subsequent chapters.

1.3 Organization of the thesis

This thesis is organized as follows;Chapter 2, titled compressive sensing based super-resolution proposes new methods of evaluating sparse recovery and reconstruction. Analyses and evaluation of trained and randomly sampled dictionaries is performed and their implications on incoherence and sparsity is noted.In addition to providing these analyses, some of the general properties of trained dictionaries for different super-resolving capabilities is discussed. Chapter 3 is titled affine sparse codes for image classification. This chapter proposes and evaluates novel methodologies of feature extraction,feature formation through sparse representation and dictionary learning.Experimental results and benchmarking with the most recent techniques is detailed.Chapter 4 discusses on the newest technique in image denoising through matrix completion. This method proposes and evaluates using sparse representation along with singular value thresholding techniques to search for the best denoised patch. Results are evaluated with the state of the art techniques and along with the effectiveness of the method. The thesis concludes with Chapter 5 detailing on the conclusions and future work.

Chapter 2

SUPER-RESOLUTION THROUGH COMPRESSIVE SENSING

Super-resolution(SR) is the process of combining multiple low resolution images to form a higher resolution one.Usually it is assumed that there is some (small) relative motion between the camera and the scene, however motionless super-resolution is indeed possible if other imaging parameters (such as the amount of defocus blur) vary instead [34]. If there is relative motion between the camera and the scene, then the first step to super-resolution is to register or align the images; i.e. compute the motion of pixels from one image to the others. However this may not be the only form of Super-resolution, since there might be a need of super-resolving from single image. Then we would not be able to use data from multiple images to obtain a better high resolution version of the input image.Super-resolution from a single image has received much attention with the advent of Compressive Sensing(CS). There also have been other methods which have been successfully able to achieve good results for different super-resolving factors [25]. One such method utilizes the patch redundancy across the same scale and different scales. The approach is based on the observation that patches in a natural image tend to redundantly recur many times inside the image, both within the same scale, as well as across different scales. First an overview of some of the previous methods is investigated followed by a detailed description of the proposed method.

2.1 Super Resolution: An Overview

SR methods have been broadly classified into two families of methods namely: (i) Classical multi-image SR and (ii)Example based SR.In classical multi-image SR a set of low resolution images of the same scene are taken (at subpixel misalignments). Each low resolution image imposes a set of linear constraints on the unknown high resolution intensity values. If enough low-resolution images

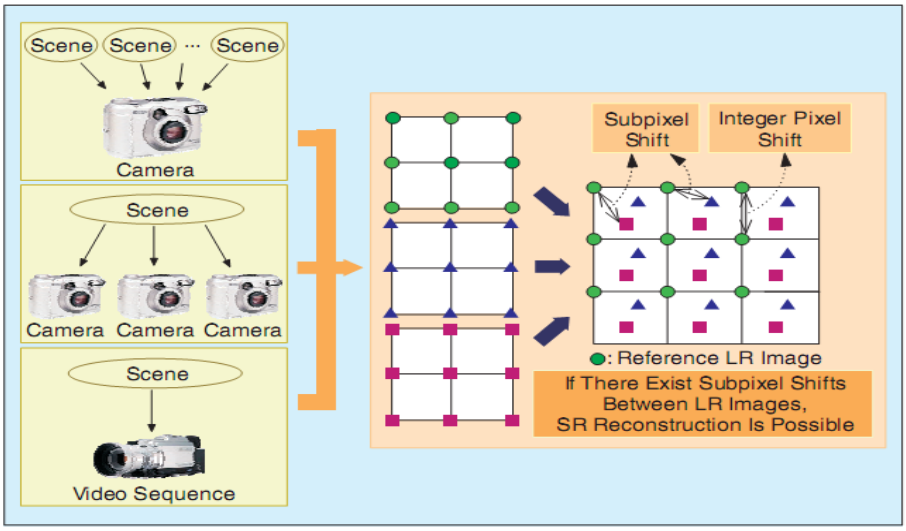

Figure 2.1: Basic Premise of Classical Image SR.

are available (at subpixel shifts), then the set of equations becomes determined and can be solved to recover the high-resolution image. Practically, however, this approach is numerically limited only to small increases in resolution (by factors smaller than 2) [22],[35],[36],[37]. Fig. 2.1 shows a typical classical image SR framework. Now the next step would be to obtain the SR image from multiple low resolution (LR) images. Multiple LR images of the same scene are a basic necessity for increasing the spatial resolution in SR techniques.The LR images are subsampled (aliased, no low pass filtering) as well as shifted at subpixel resolutions. Shifting by an integer amount in the LR image results in the same information and it would not add any new information for reconstructing the HR image. But LR images with different subpixel level shifts may add new information and are useful in constructing a HR image even if they have aliasing present in them. In this case, the new information contained in each LR image can be exploited to obtain an HR image. But in order to achieve this, multiple images with relative motion between them should be obtained. Multiple scenes can be obtained from one camera with several captures or from multiple cameras located

in different positions or multiplse scene motions [28]. If these scene motions can be estimated within subpixel accuracy and if we combine these LR images, SR image reconstruction is possible as illustrated in Fig. 2.1. But as previously mentioned the upfactors or the super resolving factors obtained are very small. Thus these limitations have led to the development of example based or learning based SR.

2.1.1 Example based SR

In example-based SR, correspondences between low and high resolution image patches are learned from a database of low and high resolution image pairs (usually with a relative scale factor of 2), and then applied to a new low-resolution image to recover its most likely high-resolution version [25]. By repeated application of the same process images with higher SR factors have been obtained. Example-based SR has been shown to exceed the limits of classical SR. But however this does not reflect directly into reconstructing the actual HR image since there will be generation of pseudo high resolution details. In SR (example-based as well as classical) the goal is recovery. This involves generating missing high-resolution details which are not found in any individual low-resolution images. In the classical SR, this high-frequency information is assumed to be split across multiple low-resolution images, leading to information on high resolution images in terms of sub-pixel shifts and in aliased form . In example-based SR, this missing high-resolution information is assumed to be available in the high-resolution database patches or exemplars of dictionaries, and learned from the low-res/high-res pairs of examples in the dictionaries.

2.1.2 Compressive sensing based SR

Recently, Compressive Sensing (CS) has emerged as a powerful tool for solving a class of inverse/ underdetermined problems in computer vision and image processing. In this work, we investigate the application of CS paradigms on single image

Super-Resolution (SR) problems which are considered to be the most challenging in this class. In light of recent promising results, a set of novel tools are proposed for analyzing Sparse Representation based inverse problems using redundant dictionary basis.Further, novel results establishing tighter correspondence between SR and CS are provided. As such, some gains include insights into questions concerning regularizing the solution to the underdetermined problem, like: (i) Is sparsity prior alone sufficient? (ii) What is a good dictionary? (iii) What is the practical implication of non-compliance with theoretical CS hypothesis? Unlike in other underdetermined problems that assume random down-projections, the low resolution image formation model employed in CS-based SR is a deterministic down-projection which may not necessarily satisfy some critical assumptions of CS. A further investigation on the impact of such projections in concern to the above questions is provided.

SR is an inverse problem which deals with the recovery of a high-resolution image from a single or a sequence of low-resolution images based on either specific a priori knowledge or just assumed generic notion about the imaging model. In generation of low-resolution images, the imaging process normally involves low-pass filtering followed by decimation. Since such a process results in a loss of entropy, the reconstruction problem is highly underdetermined. Hence proper regularization is necessary for finding an appropriate solution, especially under large magnification factors, due to the large size of the solution space. Generic edge smoothness priors and/or other visual features are typically utilized to regularize the solution. Such examples include gradient prior [1] soft-edge prior [2], Markov Random Field (MRF) [13], primal sketch prior [23], directional-priors [20] and Total Variation (TV) [3]. The essence of these priors is to ensure coherence in the local properties of the reconstructed image. Also many algorithms extract local features and learn the local properties via recognition based priors to obtain

an appropriate high resolution image [22],[26]. Recognition and learning based super resolution algorithms [22], [24] estimate the bounds on the super resolving factor that can be carried out on natural images. Single image SR algorithms have been studied utilizing the patch repetitions across the same scale and multiple scales in natural images [25]. Sparse derivative priors, learning based image up scaling, local correlation based super resolution and survey of different techniques used in super resolution have been compared by Ouwerkerk and can be found in [27]. In all SR problems, a fundamental global reconstruction constraint is that the super-resolved image should yield the original low-resolution version when the assumed imaging model is applied. The Iterative Back-Projection is one such method widely employed for this purpose [6] [7].

The recently-emerged idea of Compressive Sensing (CS) theory provides a different perspective in solving large underdetermined problems, exploiting sparsity as a prior [15] [16] [17] [18] [21]. This powerful and promising tool has proven to be effective for a wide range of problems of this class, including sub-Nyquist sensing of signals and coding, image denoising, and de-blurring [11] [15] [16]. Very recently, [7] addressed the SR problem using a sparse representation-based algorithm, reporting superior results. However, some fundamental questions are yet to be answered, such as: whether CS paradigms can address SR problems? Is the theoretical hypothesis of CS satisfied in the case of SR problems, and what are its implications in practice? In this study, our goal is to holistically understand and answer how effective are CS paradigms with respect to the SR problem. Since CS has emerged as a powerful tool, it is of great interest and importance to address the fundamental questions in CS for underdetermined problems like SR. Here, we seek to understand and establish a relationship between CS and SR theories and provide a better understanding of the role of sparsity priors and the properties of the projection operator and dictionaries. In this study, the goal is to holistically

understand and answer how effective are CS paradigms with respect to the SR problem. Since CS has emerged as a powerful tool, it is of great interest and importance to address the fundamental questions in CS for underdetermined problems like SR. Here,an attempt towards understanding and establishing a relationship between CS and SR theories and provide a better understanding of the role of sparsity priors and the properties of the projection operator and dictionaries is undertaken.

2.2 SR in a CS framework

For completeness, we first briefly review some necessary background about the CS. Suppose a signal $x \in \Re^N$ is S-sparse with respect to a basis $\Psi \in \Re^{NxN}$(i.e, x = $\Psi\alpha$, $\| \alpha \|_0 = S < N$) we define its measurement as $y = \Phi x$, $y \in \Re^M$, using the projection operator $\Phi \in \Re^{MxN}$,M<N. Then, CS says that x can be recovered from $y \in \Re^{MxN}$ using a decoder Δ that in involves solving either of the following l_1 minimization problems.

$$B.P.\ \hat{\alpha} = argmin \| \alpha \|_1 \ s.t. y = \Phi\Psi\alpha \tag{2.1}$$

$$B.P.D.N.\ \hat{\alpha} = argmin \| \alpha \|_1 \ s.t. \| y - \Phi\Psi\alpha \| < \epsilon \tag{2.2}$$

Eq.2.1 is Basis Pursuit (BP) and Eq.2.2 is the Basis Pursuit De-noising (BPDN) approach [15]. Faithful signal recovery is guaranteed by any decoder Δ provided $M \geq C\mu^2(\Phi, \Psi)SlogN$, where C is a constant and $\mu(\Phi, \Psi)$ is the coherence between the pair of the measurement matrix and the sparsifying basis Φ, Ψ [15],[16],[17],[18] and S being the sparsity of signal x and N being the dimension of signal x. Here coherence is defined as,

$$\mu(\Phi, \Psi) = \max_{j,k} |\langle \phi_j, \psi_k \rangle|, \phi_j \in \Phi, \psi_k \in \Psi \tag{2.3}$$

For the lowest number of measurements M to be taken for an optimal S-sparse signal what is the best measurement matrix $\Phi \in \Re^{MxN}$?The answer is provided

by the notion of Restricted Isometry Property (RIP) by Candes [15],[16],[17]. RIP of order S is satisfied by $\Phi\Psi$ with a constant $\delta \in (0,1)$ if

$$(1-\delta) \parallel x \parallel_2^2 \leq \parallel \Phi\Psi\alpha \parallel_2^2 \leq (1+\delta) \parallel x \parallel_2^2, x \in \Sigma_S \tag{2.4}$$

Here Σ_S is the set of all S-sparse vectors x, $(x = \Psi\alpha)$. A reconstruction of x is possible from $y = \Phi x$ using a CS decoder Δ under the condition that $\Phi\Psi$ satisfies the RIP property of order 3S for some $\delta \in (0,1)$.The error bound is given by:

$$\parallel x - \Delta(\Phi\Psi\alpha) \parallel_2 \leq C\sigma_s(x)_1/\sqrt{S} \tag{2.5}$$

Here $\sigma_s(x)_1 := \inf \parallel x - z \parallel_1$, z $\in \Sigma_S$ is the error of the S-term approximation to x in l_1 norm. For optimal reconstruction results,$\Phi\Psi$ has to satisfy RIP of order S given by [14],[15],

$$S = M/log(N/M) \tag{2.6}$$

Another notion says that if the sparsity is bounded as

$$S \leq M/(C\mu^2(\Phi,\Psi)log(N/\delta)) \tag{2.7}$$

for a given coherence $\mu(\Phi,\Psi)$ and a constant δ, then a decoder Δ can perfectly recover x with probability exceeding 1- δ. Thus, for a given pair Φ, Ψ , higher the RIP (order S) (or equivalently lower the coherence $\mu(\Phi,\Psi)$, better the reconstruction (i.e., better reconstruction guarantee and smaller reconstruction error) for any decoder Δ. In most CS problems, the basis Ψ is generally assumed to be orthonormal (ONB), and the projection Φ is usually chosen as a random Gaussian matrix as it possesses good RIP and is highly incoherent with most Ψ [15]. With the above knowledge we can map an SR problem in a similar way.We can consider y to be a low-resolution image and x being the high-resolution image and the projection matrix Φ may be a deterministic imaging model and the sparsifying basis Ψ may not necessarily be an ONB but an Arbitrary Redundant Dictionary (ARB) (denoted as D$\in \Re^{NxK}$,K >>N).

2.2.1 Problem Definition: SR based on CS

Before we address the questions on SR projection operator, dictionaries and CS solvers, it is necessary to formally formulate the SR problem in a CS framework. The SR problem is to recover the high-resolution image X back from a single or multiple low-resolution images Y_i,i=1,..J. In this analysis, we consider only the case of a single input image (J=1). The low-resolution image Y is obtained from the high-resolution image X, through the following image generation model,

$$Y = RL_pX = LX, X \in \Re^{PxQ}, Y \in \Re^{\tilde{P}x\tilde{Q}} \tag{2.8}$$

where L_p is generally a low-pass operator and R is a decimation operator that does the downward sampling of X. And U= $P/\tilde{P}(=Q/\tilde{Q})$ is the decimation factor, and we will call it simply the up-factor. The entire operation is linear in nature and we represent it as a matrix operation L=RL_p. Since Eq.2.8 results in information loss, it is a challenging process of recovering the original image through the inverse operation.Instead of solving the recovery problem for an entire image, the problem can be split into number of small parts which we call the patch which is used to recover original patch [7] with an additional constraint that the final image obtained should result in an input Y when the model of Eq.2.8 is applied. Now, if $x \in \Re^N$ is a 1-D representation of a small patch of X, we have an over-complete dictionary D $\in \Re^{(NK)}$ that can sparsely represent x as,

$$x = D\alpha, \| \alpha \|_0 = S, S < K \tag{2.9}$$

then, the low-resolution patch is given by,

$$y = Lx \tag{2.10}$$

where x is projected using the low pass operator to obtain y , similar to a CS measurement. Certain CS recovery conditions are to be satisfied Eq. 2.6,Eq.2.7,

if the sparse vector α in Eq.2.9 can be recovered from the lower dimensional measurement.

$$y = LD\alpha \tag{2.11}$$

Eq.2.11 is an optimization problem which can be solved either by Eq.2.1 or Eq.2.2. A necessary condition to make sure that the final solution complies with the imaging model Eq.2.7 [7] is to apply a global reconstruction constraint like back-projection. In the next section, we present theoretical analysis of the projection-operators and the dictionary.

2.2.2 Theoretical analysis of projection operator and redundant dictionaries

Our goal is to evaluate and understand the nature of a given pair of projection-operator and dictionary, (L,D) in the context of SR and compare it with (Φ,D). Again, we emphasize L is a deterministic projection operator and Φ is a random projection operator and D is an overcomplete dictionary (ARB). Most of the CS theories have been developed for sparse representations on ONBs, but recently in [8], [19] attempts have been made to generalize these theoretical results on sparsity/recovery constraints to any ARBs. For example, mutual-coherence μ of 2.3 is a good measure and can be relied on for evaluating tighter sparsity bounds of a CS system with (Φ,Ψ) (ONBs). So we will resort to theoretical analysis of the properties of the projection operator and the redundant dictionaries to understand the sparsity bounds and its relation to the mutual coherence.

A. The L Projection Operator

An important property about L operator is its deterministic nature and frequency discriminative nature since it preserves only the low pass information of the signal x. Since L exhibits good RIP characteristics (CS property)(Eq.2.4) it can be represented in the matrix form which is circulant in nature and it also satisfies the property,$l_{i+1,j+u}$=$l_{i,j}$, where u:= N/M and i,j are row, column indices incremented in modulo N arithmetic. While Φ is not frequency discriminative, it also preserves

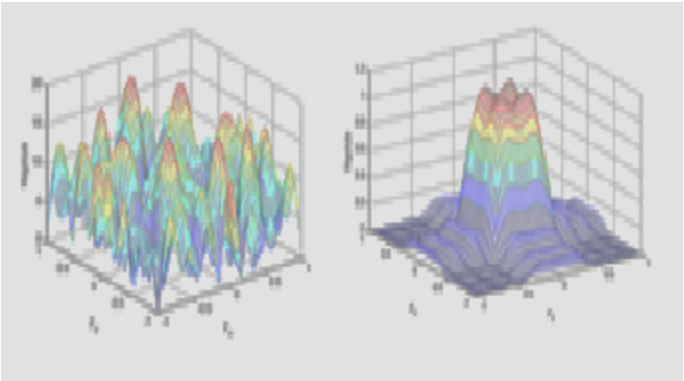

Figure 2.2: 2-D frequency response of a random projector (left) and a L.

all frequencies of a signal when subjected to a Φ operator.Fig. 2.2 visualizes the 2-D frequency responses of the two operators. In this regard, we draw an interesting connection to the results in [14] obtained for deterministic CS matrices.

Specifically, Theorem 3.4 of [14] states that the circulant matrix constructed from finite fields satisfies the RIP property of order S given by

$$S < \sqrt{M}.log(M/8.log(N/M)) \tag{2.12}$$

and since L$\in \Re^{MN}$ is a similar matrix, we may use Eq.(1.13) as an upper-bound. Hypothetically, either considering L independently or in conjunction with an ideal-basis, Eq.1.13 indicates a much inferior bound on sparsity compared to the case of random operator required for optimal reconstruction. For example, if we consider an ideal basis and an imaging model L, image patch y of M=9 (3x3 pixels), and an original x patch of N=81 (9x9), then, the upper bound for sparsity is S<1.4 or S=1 as opposed to S<9 for a random operator Φ (Eq.2.6). The upper bound on S =1 confirms the fact that the image patch itself has to be the basis. But in reality, such basis may not exist and we may need to resort to dictionaries D. Thus, the sparsity bounds should be evaluated using joint properties of pair (L,D).

B. Redundant Dictionaries in SR

What is a good dictionary? This is the fundamental question which has been re-

searched on in the recent years for various scenarios or goals (sparse-representation /coding, recognition etc). A very naive over-complete dictionary is one whose base-atoms are the element-type itself selected from random sampling of some training data. In case of SR they are raw-image patches (these are simply called random-sampled or RS). We have also seen recent attention on training algorithms with a goal to obtain compact dictionaries [10],[11],[21]. In SR, the goal is not sparse representation, but sparse recovery. In this section, our objective is to gain insight on properties and performance of RS and trained dictionaries. Unlike in ONBs, which provide a unique sparse representation, the first question is if unique single sparsest representation exists for a system Eq.2.9. According to [19], if the condition

$$\| \alpha \|_0 = S < 0.5(1 + \frac{1}{\mu D}) \tag{2.13}$$

is satisfied, then the sparse-representation α is unique and also the sparsest. From a low-dimensional space with L, perfect-sparse-recovery of α requires much stricter criterion to be satisfied,

$$\| \alpha \|_0 = S < 0.5(1 + \frac{1}{\mu LD}) \tag{2.14}$$

In practice, for most D (RS or well-trained), μD and μLD are almost close to 1 (see Fig.2.3), which yields sparsity bounds no better than S=1. Thus, theoretically, this means that optimal recovery is possible only if there is exactly one match in the dictionary. Thus far, we may say this is an over-pessimistic demand that does not give us understanding of aforementioned questions on the (L,D) pair and different types of D.

C. Proposed Tools for Analysis of L, Φ and D

As we can see from the above analysis, there is a need to evaluate the joint properties of the (L, D) pair and also the mutual coherence evaluated for different dictionaries may not provide complete information on their properties. Similarly

Patch =M	*Up-Factor*	*μ(LD)*	*μ(ΦD)*
9x9	1	0.9448	
6x6	1.5	0.964	0.948
4x4	2.25	0.982	0.953
3x3	3	0.995	0.988

Figure 2.3: Grammian of D, LD, ΦD for original dictionary D $\in \Re^{81x1024}$(9x9)trained by [21].

M	D Type	*GramH*, p=2 : histogram as % age			
		0-0.1 (Best)	0.1-0.3 (Good)	0.3-0.8 (Mid)	0.8-1 (Worst)
9x9	*D*	50.73	38.31	10.27	0.05
3x3	*LD*	17.53	32.60	35.88	2.5
	ΦD	19.36	34.99	35.11	1.3

Figure 2.4: GramH of D, LD, ΦD (p=2,4 bins)for original dictionary D $\in \Re^{81x1024}$ (9x9)trained by [21].

for an ARB (D), complete reliance on for a stricter sparsity bound will always be misleading, since D$\in \Re^{(NK)}$ has K>>N. So, one may obtain similar μ for a relatively well-conditioned D having fewer similar-atoms as well as a totally ill-conditioned one with large number of similar atoms. Other options may include relying on RIP-based on uniform uncertainly principle (UUP) [18]. Reasoning similar to the case of coherence, RIP constants only give the worst case conditioning of the dictionary, so are not completely reliable. Another notion is a geometrical view point in [17]. Since none of the measures described above provide a clear description of the properties of the dictionaries, there is a need for new method of analysis which provides insights into the nature of the dictionary and its atoms and its collective influence on signal reconstruction. In this thesis, in addition to coherence, we propose new methods to evaluate dictionary D or its

projection OD matrix, (O is L or Φ), based on the Gram-matrix which is defined as, G(D)=$\tilde{D}^T\tilde{D}$ where $\tilde{D}_i = D_i/ \parallel D_i \parallel_2$ (i.e., columns normalized by l_2 energy). Then, the coherence (μ) of the dictionary is redefined as,

$$\mu(D) \cong max_{1<i,j<K;i\neq j} G(i,j) \tag{2.15}$$

and takes the values in [0,1]. A 0 signifies least coherence (orthogonal column vectors) and a 1 means highest (exactness). In the rest of the thesis, we will resort to the following new statistics for analyzing D or OD (O $\cong$ L orΦ). (i) Gram-Histogram: This is a histogram of μ defined as

$$GramH\ D, K, p \cong hist(\mu(D_p)), bins \in [0,1] \tag{2.16}$$

where D_p is the set of all sub-matrices of D formed by choosing p column support from the set 1.K. There are ${}^KC_p$ such possible elements. Thus, this is similar to RIP evaluation, but additionally it provides statistics as to how well-conditioned the base atoms are. For example, if p=2, then (16) evaluates the distribution of coherence for all ${}^KC_p$ pair-wise combinations of base-atoms. This can be evaluated over B bins in the range [0,1]. More entries in the lower bins (near 0) means that on a pair-wise basis, most atoms are highly uncorrelated. More entries near 1 signify that many atoms are similar (ill-conditioned). If evaluated for OD, it gives joint properties of (O,D). For p=2, (16) can be easily implemented by simply plotting the histogram of Gram matrix G with diagonal explicitly made, say -1 ($\notin$[0,1]). (ii) Gram-Member: This is another metric defined as

$$GramM\ D, T, p, B \in [0,1] \cong \tilde{K} \tag{2.17}$$

Here T$\leq$K is a threshold, B is a bin in the range [0,1]. $\tilde{K} \leq$K gives the number of Gram members for bin B. The i-th base atom (column vector) D_i is called the Gram-member of bin B under threshold T, if the following is true: one can find at least T sub-matrices in the set D_p involving D_i, for which $\mu(D_p) \in$ B.

To explain this better, let us take an example of p =2, T=50 and dictionary D of size K=1024. Now D_p is the set of all pair-wise combination of sub-matrices and there are 1023 such pairs for a base-atom D_i denoted as $D_{p,i}$. If there are at least 50 (T) elements in$D_{p,i}$for which, $\mu(D_{p,i}) \in B$, then, we declare D_i to be a member of bin B.We repeat this for allD_i,i=1..K. The final result of GramM of is simply the count of the number of members in bin B. Thus, if B near zero, [0,δ) (for a small δ), GramM would provide the information about the number of base-atoms, that maintain ultra-low correlation with atleast T other base atoms. Greater this number, better it is. Similarly, for a B near one [1-δ,1] , GramM should be as low as possible. Note that greater the T, stricter is the measure. If the percentage of base atoms with ultra low correlation with atleast T other base atoms is close to 100% then the dictionary exhibits excellent well-conditionedness. GramM conveys more local information since it provides information regarding the uncorrelatedness between the base atoms and GramH provides global information on well-conditionedness of the dictionary D or the pair (O,D) as a whole. In our analysis, we typically use p=2 and classify the coherence bins as [0, 0.1] (best), (0.1, 0.3] (good), (0.3,0.8] (mid) and (0.8,1] (worst) for analysis of GramH. Also, in case of GramM, we simply use bins in steps of 0.1. The measure of well-conditionedness of a dictionary directly translates to a significant quality improvement in the recovered/reconstructed image. With the theoretical analysis in mind and the new tool set proposed, we now proceed to the experiments section. This includes the experimental evaluation of projection operator and dictionaries in terms of the coherence measures like GramM and GramH, and visual results to corroborate the experimental evaluation .

2.3 Experiments and Discussion

A. Evaluation and Gram Statistics Validation of the L Projection operator

We consider an over-complete dictionary D with 90,000 atoms obtained by ran-

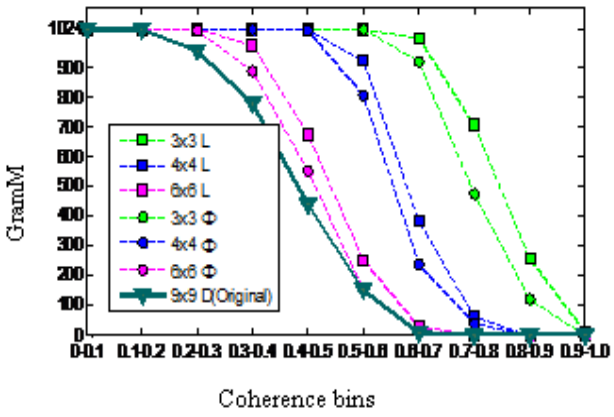

Figure 2.5: GramM:LD and ΦD for various dimensions: 3x3, 4x4 and 6x6 and original D with p=2 and T=30.

domly sampling the raw-image patches from training images. This random sampled dictionary is trained using the Feature sign search (FSS) algorithm in [21] to obtain a dictionary of size 1024. The Grammian (coherence) for D (9x9 patch size), LD and ΦD are shown in table of Fig.2.3. Clearly, this worst case RIP/coherence measure is high for all cases and shows only marginal superiority for Φ. Thus, we resort to Gram-statistics measures described earlier. In Fig 2.4, GramH measures (with p=2) are compared for (L,D) and (Φ,D) pair for M=9 from original N=81. Clearly, D is far well-conditioned: 50 In Fig 2.5, we present the GramM measures for (L,D) and (Φ,D) for various projection dimensions (3x3,4x4,6x6), evaluated with p=2 and T=30. For a fixed up-factor, Φ curves are superior compared to L (higher coherence bins have lesser Gram-members for Φ than L).This trend is true for any up-factor. On the other hand, compared to D, both LD and ΦD degrade as up-factor increases. Thus, in line with the theoretical results, these measures also show that, from a CS connection L is inferior compared to Φ. Performance Evaluation: With this, we are now interested in understanding the practical implications of L in SR. We evaluate the performance by devising experiments determining the distortion characteristics in super-resolving

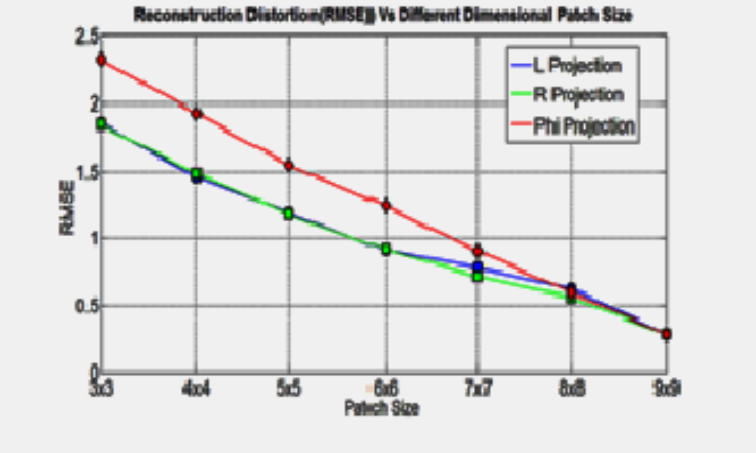

Figure 2.6: RMSE reconstruction curves L,Φ and R for various up-factors

image patches by different up-factors (Us). We selected multiple 9x9 patches x_i with varied texture information from multiple high-resolution test images. The corresponding low-resolution patches y_i were created assuming L to be a Gaussian blurring kernel with cut-off frequency π/U, followed by a decimation U↓ (or R). We recover the original patch by solving for α in 2.2 using BPDN 2.11. Fig. 2.6 shows the results of the experiment average RMSE curves for L and Φ operators for various up-factors. Although the Φ does not have any semantic meaning in SR, we use it to benchmark and understand L for reasons discussed earlier. From theoretical perspective and Gram-analysis, as expected ΦD is better conditioned than LD. However, this does not translate to superior performance as indicated in Fig.2.7. In fact, from Fig.2.6, the L curve is better than Φ, especially for dimensions lower than M=7x7. An intuitive reason is provided for this contradiction explaining two cases which result from this. Since the patches of x_i of natural images do not occupy full nyquist range, we can say that x is band-limited to say π/W, for some W=1(π being the Nyquist frequency).Suppose y $\in \Re^{MM}$ and x $\in \Re^{NN}$ and U = N/M, we have the following two cases:

- U>W. Assuming good transition characteristics, L preserves most of the

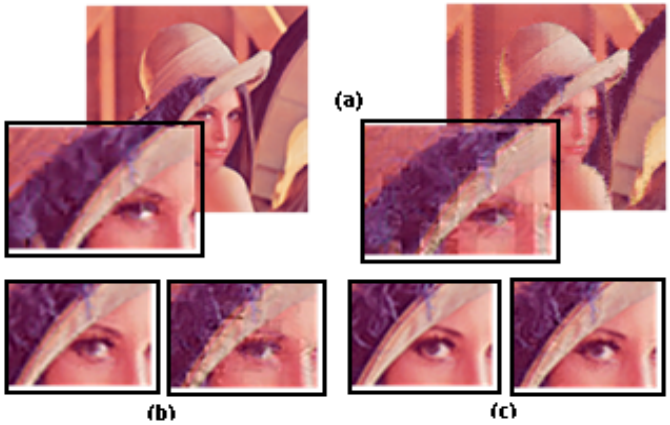

Figure 2.7: Visual image results (a) Reconstructed 9x9 patches from 3x3 for L(left) and Φ (right).Reconstruction of 9x9 from (b) 5x5 for L (left) Φ (right) and (c) 6x6 dimensions for L (left) Φ (right)

energy of the signal x is preserved in only M points occupying frequencies (0, π/U). All the information in the range (π/U, π/W) is lost. While ϕ preserves all the information (0,π) in M points, L does not waste any measurements capturing frequencies above π/U. Thus when U is increased the RMSE of estimated $\tilde{x}$ w.r.t x is much superior for L than ϕ, in the range (0, π/U), leading to better overall-RMSE. Since the problem being dealt with here is undetermined, recovering all frequencies seem to be much more harder than recovering only frequencies which L has not captured which is much lesser than what ϕ has captured.

- U<W, then the probability of perfect recovery for L is high. Visual results in super-resolving Lenna image for U=3 is presented in (a) of Fig.2.7, which corroborate these facts. The left image is for L and the right for ϕ. Also see (b) of Fig.2.7 and (c) of Fig.2.7 showing high-texture section (recovery) for other up-factors. This is in line with the fact that L preserves all of the energy within U while Φ tries to preserve all of the energy within W. As the up-factor increases Φ closes in on L. Now that the properties of L are

evaluated experimentally, we focus the next subsection on the properties of D in SR.

B. Experimental evaluation of redundant dictionaries

We resort to Gram statistics for evaluation of dictionaries. The high patch dictionary D and low patch dictionary LD are evaluated for trained dictionaries like the feature sign search (FSS), KSVD and non-trained dictionary like the randomly sampled (RS) dictionary.

Gram Statistics Validation: We consider the two categories of Ds of size 1024 for N=81 (9x9) high-resolution patches (i) RS (evaluated for various trials of random sampling). (ii) Two examples of trained dictionaries: Feature-Sign-Search (FSS) [21] and K-SVD [10], [11]. Fig. 2.8 provides the GramH measures for p=2 and four bin ranges for these types of Ds and their low dimensional versions LD. Clearly, for the lower coherence bin (0-0.1) in D, the statistics indicate that training reduces the correlation among base-atoms. FSS is overall better conditioned than KSVD with 50% against 38% of pair-wise correlations respectively, while RS has 30% in the (0-0.1) region. On the other hand, the worst case correlations in the region (0.8-1) of FSS is very low (0.05%), but significant (0.33%) for RS. KSVD dictionary has higher value in this bin compared to RS. The general conditioning of LD for all types of D degrade (see Fig 2.8). For a 3x3, the numbers maintain similar trends across FSS, KSVD and RS dictionaries. The number of worst-case correlations increases to a quite high of 6.5% in RS, while for the trained they remain relatively low. Fig. 2.9 compares the GramM measures of FSS and RS (D and LD for 6x6 and 3x3). Clearly, the curves indicate that FSS has far superior conditioning than RS both in high and low-resolutions dictionaries **D** and **LD**.

M	D Type	*GramH*, p=2 : hist as % percentage			
		0-0.1	0.1-0.3	0.3-0.8	0.8-1
D (9x9)	FSS	50.73	38.31	10.27	0.05
	KSVD	37.7	36.97	24.34	0.97
	RS	29.85	42.87	26.94	0.33
LD (3x3)	FSS	17.53	32.60	35.88	2.5
	KSVD	17.11	31.60	47.47	3.79
	RS	14.28	27.34	52.28	6.07

Figure 2.8: GramH (p=2, 4 bins) for various types of dictionaries D of length 1024 (9x9 patches) and two categories: RS (average for various RS), and trained (FSS and KSVD). Also presents GramH for LD (3x3)

C. SR: Solution Space and CS Solvers

We gained insights on the role and properties of dictionaries in SR in the previous sub-section. This sub-section bridges understanding of some important questions related to sparse solution and recovery in SR: (i) The role and constraints on sparsity; (ii) Solution space and CS solver; (iii) Is uniform sparse-recovery possible or important? This section reviews some of the preliminary experiments conducted previously by [38]. Also a modified results of their experiments are presented and analyzed in this section. Theoretical and Practical Connections: For a dictionary satisfying 2.13, the BP problem 2.1 is guaranteed to find the unique sparsest solution [19]. However, for actual SR dictionaries discussed in the previous section, a BP solver like l_1-magic has stability issues due to the size and poor conditioning properties of the dictionaries (compared to ONBs) In practice, the unconstrained version of BPDN 2.2 cast as the following

$$\tilde{\alpha} = argmin\ \tau \| \alpha \|_1 + 0.5 \| y - \Phi\Psi\alpha \|_2^2 \tag{2.18}$$

is a suitable choice for the CS decoder. Here τ is a regulariser that controls the tradeoff between sparsity and fidelity. In this subsection, we study and provide

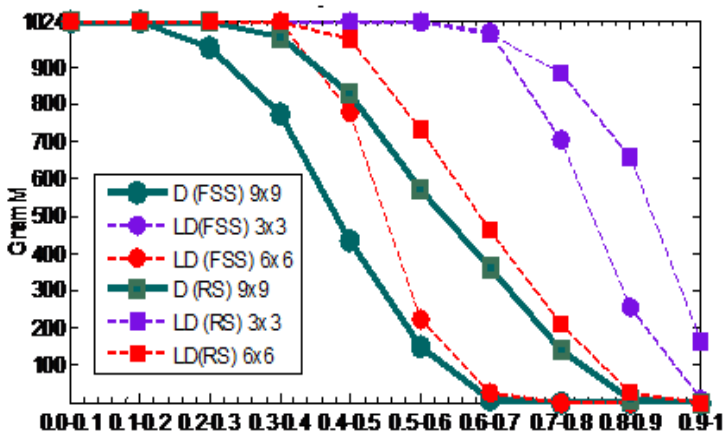

Figure 2.9: GramM (17) with p=2 and T=30 for FSS and RS.

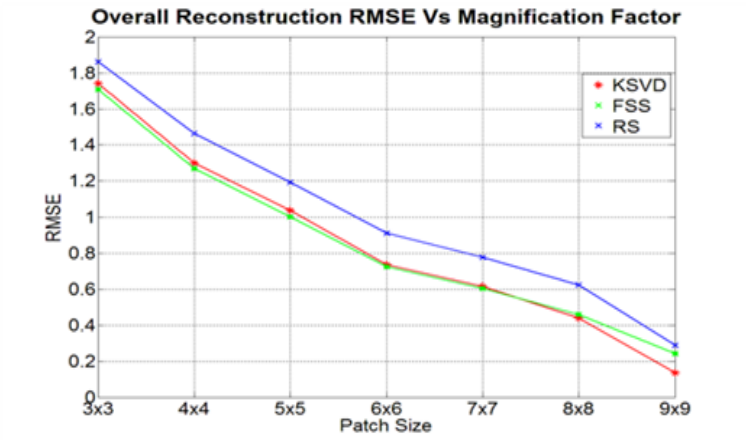

Figure 2.10: RMSE performance curves for various up-factors for the Random sampled(RS) and trained(FSS,KSVD) dictionaries. This curve is an average evaluated over various patches. Clearly, FSS and KSVD dictionaries perform better than RS.

interesting insights on the question how necessary is a sparse solution for SR? and What is a suitable value for the τ? Accordingly we are interested in the following zones of operation [19], based on t, when solving 2.18

- (i)For τ=0, (2.18) reduces to an l_2 problem.
- (ii)For τ=0^+ (positive but arbitrarily close to 0), the unique optimum point of (2.1) or BP coincides with (2.2) or BPDN under certain conditions [19].

- (iii)For τ in the interval, $(0^+,\tau_{max})$where, τ_{max}=$\parallel (LD)^T y \parallel_\infty$ the solutions of BP and BPDN diverge.

As τ increases, the sparsity of the solution improves at the cost of fidelity. We want to study the behavior of the CS solver in various ranges of the regularizer t in terms of sparsity and we define these important aspects on sparse solution: *Uniform Sparse-Representation:*Sparse representation problem is defined in 2.9 is recalled here,

$$x = D\alpha_H, \parallel \alpha_H \parallel_0 = S, S < N \tag{2.19}$$

and the sparse analysis is achieved by solving 2.18. We represent τ values of 2.18 as follows,

$$\tau = \beta \parallel (D)^T x \parallel_\infty, \beta \in (0^+, 1) \tag{2.20}$$

With τ value set with a β as per 2.20, we call the sparse coefficient recovered by 2.18 as $\alpha_{H\beta}$.The support of $\alpha_{H\beta}$ is a set T_β of size S_β chosen from 1K (where K is length of D). We say that the BPDN decoder performs uniform sparse representation, if $T_{\beta 1}$ is a subset of $T_{\beta 0}$ with $S_{\beta 1} \leq S_{\beta 0}$, for any $\beta_1 > \beta_0$. This is the same best Sterm sparse approximation observed as β or τ increased. *Uniform Sparse-Recovery:* In SR what is important is sparse recovery. This involves (see 2.10), solving for α_L

$$y = LD\alpha_L, \parallel \alpha_L \parallel_0 = S_L, S_L < K \tag{2.21}$$

by BPDN and form an SR patch as $\tilde{x}$ =Dα_L. The equivalent τ values in solving the system of (2.21) is again defined similar to (2.20), except that D is replaced by LD and x by y. Now uniform sparse recovery happens when the support of $\alpha_{L\beta}$ is a subset of that of $\alpha_{H\beta}$ (again best S-term approxiation). We are interested in analyzing such aspects to understand the sparse SR solution space.

D. Operational Characteristics in SR

First, we perform an experiment to show the optimal zones of operation for an

acceptable reconstruction in SR.For an up-factor of 3 the reconstruction fidelity and the corresponding sparsity is determined(both in (2.19) and (2.21)) for various τ values for an RS dictionary. Fig.2.11 shows the related results. We find that the best zone of reconstruction is for a range of τ (from $0^+ > 0$) (shaded region Fig.2.11). Now as we can see there is hardly any change in the fidelity with changes in sparsity. Now this can be termed as Relaxed Sparsity Zone where the constraints of sparsity is of reduced significance. Similar trends are observed even with trained dictionaries and hence the plots are eliminated. Referring to the dotted curve of Fig.2.11 (sparse-representation problem of 2.19), we see that as τ increases, the RMSE degrades, while sparsity increases. The SR or sparse-recovery of (2.21) can perform no better than this dotted RMSE curve, (it acts as the lower bound). However, interestingly, in relaxed sparsity zone, for a wide range of τ, the recovery-performance (2.19), has stable and constant RMSE, indicating that striving for sparsity is not necessary or significant. A threshold is set to determine the impact of coefficients on sparsity. Hence only significant coefficients above this threshold are taken into account while plotting curves in Fig.2.11. A threshold is set to eliminate the smaller non-zero coefficients which might not strongly contribute towards sparsity. Note that the sparsity for recovery in (2.21) is higher than that for representation (2.19) as can be seen in Fig.2.11 and varies from 60 to 4-5 coefficients. On the other hand, striving for sparsity as per theoretical bounds of S=1 for optimal recovery is meaningless as the reconstruction heavily degrades. Further it was verified for $\tau = 0$ or an l_2 case, the results are not optimal either.

Next, we study the uniform sparse representation and recovery characteristics for the three dictionaries. For the former, we simply solve (2.19) for various values of τ and plot the percentage common support of $\alpha_{H\beta}$ between $T_{(\beta 0^+)}$ and T_β for all other $\beta > 0^+$. For the latter, we simply plot the percentage common

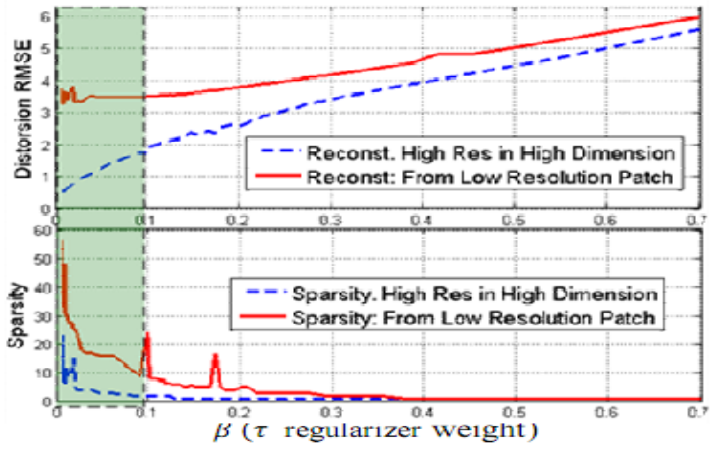

Figure 2.11: The curve shows reconstruction RMSE and sparsity as a function of β (τ) which is a fraction of the interval $[0^+, \| (LD)^T y \|_\infty]$. In the shaded zone the reconstruction is stable across all sparsity S within the range. For the other regions, even when S satisfies optimal reconstruction constraints of CS i.e.S=1, RMSE suffers.

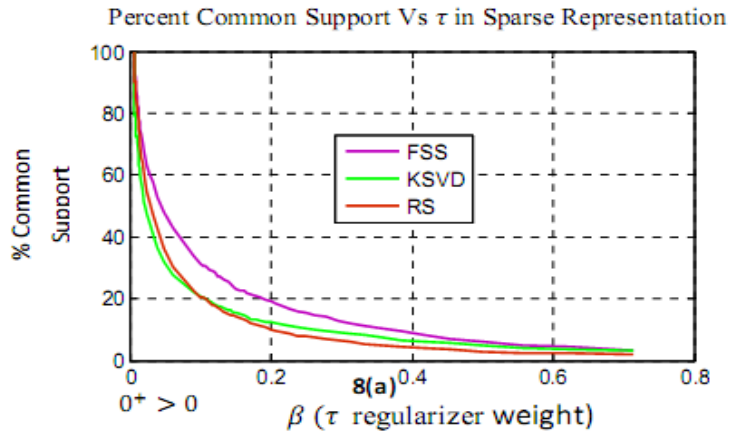

Figure 2.12: Evaluations of percentage common supports for uniform for sparse-representation

support between $\alpha_{H\beta}$ and $\alpha_{L\beta}$ as a function of $\beta(\tau)$. Again similar to the case for determining sparsity, a threshold is set and the coefficients above this threshold are used for finding indexes of common supports. Common supports are calculated as indexes of coefficients which contribute strongly towards sparsity. At different sparsity levels or at different t, common indexes with coefficient values

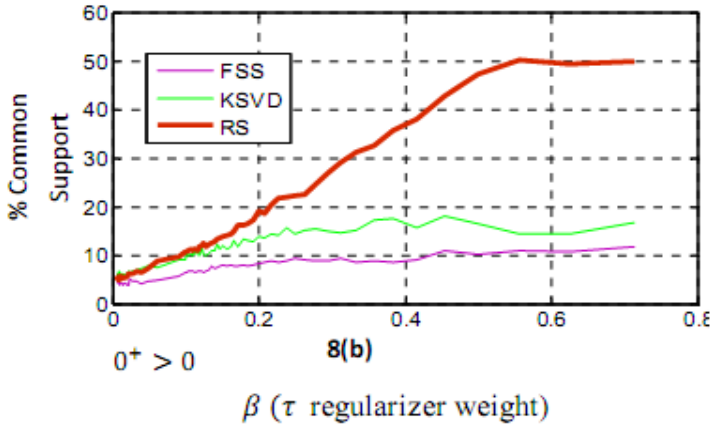

Figure 2.13: Evaluations of percentage common supports for uniform for sparse-recovery

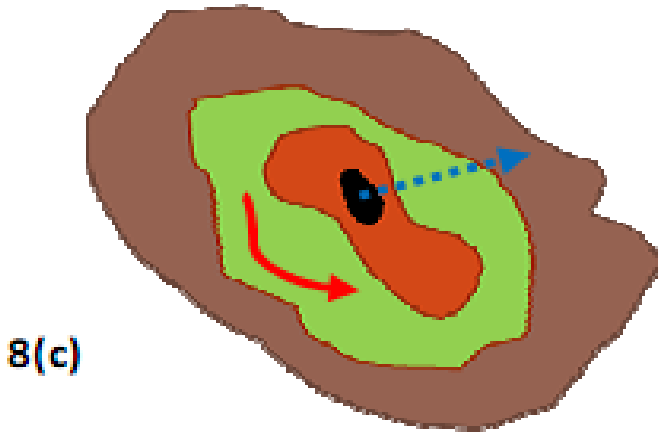

Figure 2.14: Evaluations of percentage common supports for visualization of SR solution space, with concentric regions representing relaxed sparsity zones.

above a specified threshold form the common supports $\alpha_{H\beta}$ between $T_{(\beta 0^+)}$ and T_β for the sparse reconstruction case and between $\alpha_{H\beta}$ and $\alpha_{L\beta}$ for the sparse recovery case. Our observations are as follows: (i) Uniform sparse representation is satisfied for all three dictionaries to a similar degree (see Fig.2.12). (ii) Interestingly, uniform sparse recovery characteristics are much better and consistent with increase in τ for RS (see Fig.2.13). The common support forms a monotonically increasing curve for only RS. However, despite such clean characteristics (which

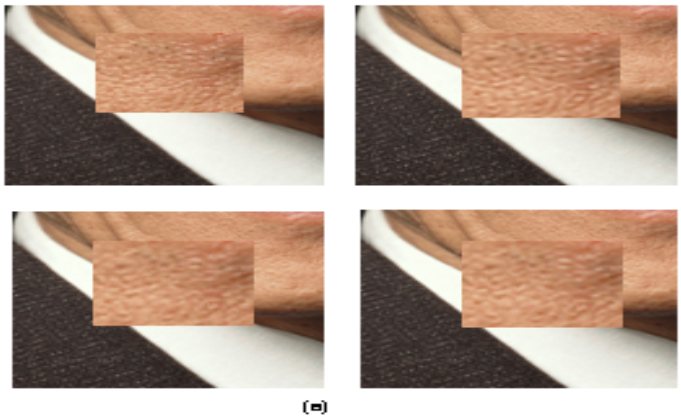

Figure 2.15: Visual Results (a) for an up-factor =3

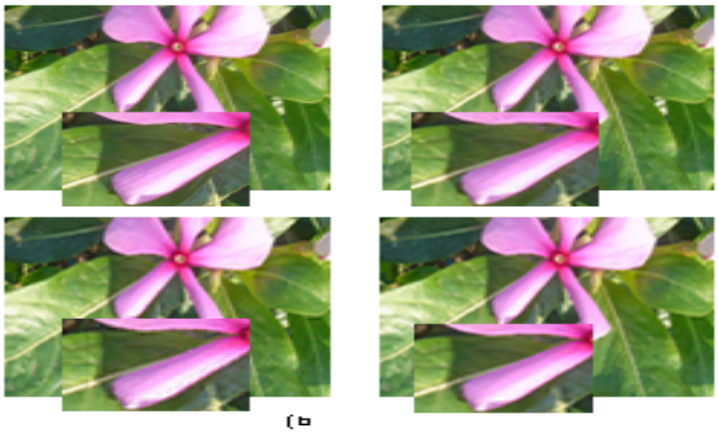

Figure 2.16: Visual Results (b) for an up-factor =3

are important in CS), we saw that RS performs inferior to trained counterparts. This along with earlier discussions on sparsity/relaxed sparsity zones corroborates the fact that in SR, uniform sparse recovery is not important and does not guarantee better results unlike in conventional CS using ONBs.

Finally, from these discussions, we visualize the solution-space in SR problems (see Fig. 2.14). As shown in Fig.2.14, it consists of concentric regions of sparse solutions yielding constant MSE also referred to as relaxed sparsity regions with sparsity being relaxed as we move outwards from central black region to

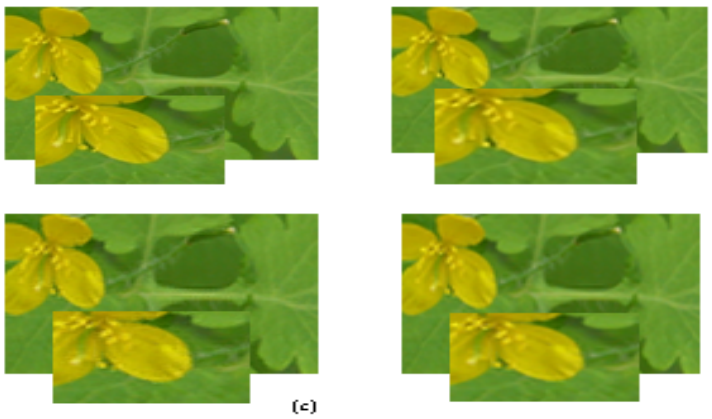

Figure 2.17: Visual Results (c) for an up-factor =3

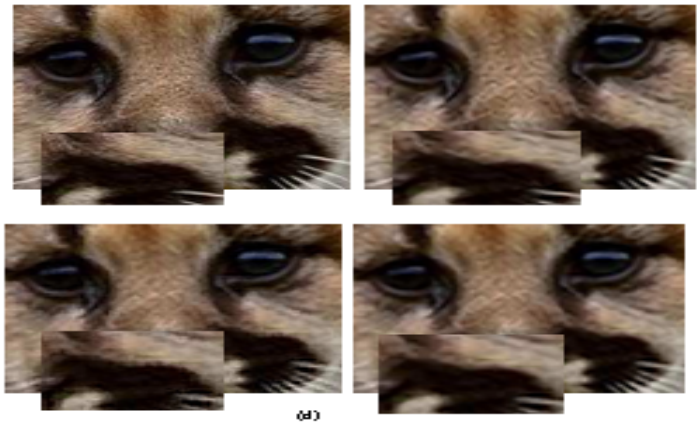

Figure 2.18: Visual Results (d) for an up-factor =3: Top left in each of (a),(b),(c),(d) is the original image. Top right in each of them is generated using Feature Sign Search(FSS) dictionary, bottom right in each of them generated using KSVD dictionary and bottom left in each of them generated using Randomly sampled(RS) dictionary. When we observe closely we can see how there is slight degradation in image quality as we move clockwise from top left to bottom left.

outer brown region. These points may have widely varied sparsities, with or without common supports (i.e. need not be best S-term subsets), but yet yield similar reconstruction. For a sparse-recovery case, on varying τ, the decoder remains in the same region as shown through red arrow in Fig. 2.14 and is not promoted to a superior MSE region. But for a sparse-representation case (2.19), the decoder

follows the blue arrow, traversing across the constant MSE regions with increase in τ.We will now discuss on the visual results obtained from random sampling(RS) and trained(FSS and K-SVD) dictionaries for a set of images.

2.4 Visual Results

We now present a set of visual results to further illustrate some of the comparisons reported in the previous discussion. Fig. 2.18 shows visual results for different images for an up-factor 3. Images have been scaled for display reasons. Clearly we can see FSS and KSVD (trained) dictionaries outperform RS(un-trained) dictionary. Some important characteristics to note are:(i) Consistency of solution in whole-image (patch neighbor) is far superior for the trained dictionary case. This is due to the fact that the probability of solver picking an unambiguous base atom from a trained dictionary (FSS, KSVD) is higher compared to that of a randomly sampled dictionary (RS). This is because of the well conditionedness of a trained dictionary in terms of its uncorrelated base atoms. Discontinuity does not appear when an overlap constraint (smoothness constraint [7]) is imposed on the solver while it picks a base atom from a trained dictionary. (ii) In RS, the result shows local patch-wise discontinuities. Although these can be reduced by applying smoothness constraints [7], RS will have artifacts which cannot be removed by any type of smoothness constraints, because of reasons explained above. (iii) As we can see from objective measurement of Fig.2.19, FSS performs slightly better than the KSVD, and both FSS and KSVD perform much better than RS dictionary. The reason can be attributed to the well conditionedness of the FSS dictionary when compared to KSVD Tests were conducted on a wide variety of images using RS, KSVD and FSS dictionaries and a few results have been presented here. The patch-wise discontinuities as can be seen in RS dictionary is because of higher percentage of correlated base atoms in the [0.8 - 1] range tabulated in

Image Name	*FSS Dictionary*	*KSVD Dictionary*	*RS Dictionary*
Gentleman Image	0.927	0.979	1.01
Flower 1 Image	1.78	1.932	2.44
Flower 2 Image	1.69	1.73	2.11
Raccoon Image	0.496	0.52	0.573

Figure 2.19: Average mean squared error over all patches for each of the images shown in Fig 9. It can be noticed that trained dictionaries (FSS and KSVD) perform better than randomly sampled (RS) dictionary.

Fig 2.8. Training reduces the percentage of correlation between base atoms and minimizes the worst case correlation [0.8 1] range of Fig 2.8. The mean squared errors were obtained for all 3 dictionaries with FSS performing slightly better than KSVD dictionary. The worst case coherence plays a key role in determining the ambiguity with which a solver picks a base atom. As we can see from Fig 2.9, the untrained (RS) dictionary has higher correlated base atoms than its trained (FSS) counterpart. This is directly seen in the mean squared values obtained in Fig 2.10 as well as Fig 2.19, which is a clear indicator of inferior dictionary as in the case of RS when compared to FSS. Also one more important observation is the convergence of mean squared errors of trained dictionaries as the patch size is increased from 3x3 to 9x9. This is due to the fact that when up-factor decreases from 3 towards 1(i.e. moving from patch size 3x3 to 9x9) the ill conditionedness in terms of the GramH measure of a trained dictionary keeps decreasing. Then the GramH of LD will approach GramH of D of Fig 2.8.So clearly the trained dictionaries are superior to untrained dictionaries in terms of their grammian properties as well as in terms of mean squared errors.

2.5 Conclusions

We investigated various issues in SR within a CS framework. A strong relationship between CS and SR was established and their underlying properties were analyzed. The study, including its discussion and experimental illustrations, serves to bridge some critical gap in knowledge of CS-based SR problems. We primarily discussed on the following aspects of the problem: (i) Implication of deterministic operator. The deterministic operator LD(joint properties of L and D) when compared with random basis like ΦD yields superior performance in terms of lower reconstruction error of high resolution image. As mentioned in previous section, this is due to the fact that LD tries to preserve all energy within the downsampled spectral range, while ΦD tries to preserve in the entire spectral range.(ii) Properties and performance of dictionaries. Trained dictionaries are effective in aiding a solver to pick an unambiguous base atom for reconstruction than the untrained counterpart. This is because of the compact nature of the trained dictionaries eventually resulting in negligible redundancy in correlation between its own base atoms as opposed to random sampled or untrained dictionaries. Thus trained dictionaries result in lower reconstruction error than untrained dictionaries. (iii) Grammian Analysis. GramM and GramH respectively bring out local and global properties of the dictionaries. These properties can be analyzed to evaluate the reconstructive capability of trained and untrained dictionaries. (iv) CS solvers and solution space, with implications on sparsity, uniform sparse recovery in SR. As we could observe from the experiments, sparsity is not a necessary criterion unlike in conventional CS methods and uniform sparse recovery may not necessarily guarantee better reconstruction results as discussed in operational characteristics in SR Obviously, these understandings will provide design guidelines in designing an SR system based on the CS framework. Specifically here we emphasize the fact that theoretical study cannot provide tighter bounds or informative conclu-

sions on sparsity in sparse recovery as opposed to those obtained in the sparse reconstruction case. Thus sparsity is not a necessary criterion unlike in formal CS methods. These analyses have also provided us with some potential future directions to explore on other aspects in SR. Since CS involves theoretical analysis on sparse representation based schemes, new techniques for analysis on sparse recovery methods in CS need to be investigated. Theoretical analysis on fundamental issues like optimal set of measurements required for sparse recovery for a given up-factor needs to be understood. This should also consider the deterministic down projection model L. We note that there are other important aspects of SR, which should be considered. These include: (i) impact of non-CS priors (e.g., feature space, directional smoothness priors etc); (ii) methods of training the dictionary explicitly considering the properties of L; and (iii) the impact of the size of the dictionary on the solution space. These will be among the future efforts that would provide more insights into the properties of dictionaries and the priors involved.

Chapter 3

IMAGE CLASSIFICATION: A NEW FRAMEWORK BASED ON AFFINE SPARSE CODES

Recent years have seen an explosion of work in the aread of object recognition [38],[51],[61],[39],[40],[41]. Several datasets have emerged as standards in the community which include Coil [42],CSAIL [43],PASCAL VOC [44],Caltech-101 [65] and Caltech 256 [49].These datasets have become progressively challenging as the datasets have consistently saturated performance. The Caltech-101 dataset consists of 9144 images of cars, motorcycles,airplanes,faces etc.The MIT-CSAIL database has more than 75000 objects labeled within 23000 images shown in a variety of environments.The PASCAL VOC has around 21,738 images with 20 classes. Caltech-256 has around 30607 images with 257 classes. Image databases are an essential element of object recognition research. They are required for learning visual object models and for testing the performance of classification, detection, and localization algorithms. Fig. 3.1 shows some of the sample images from Caltech 101 and Caltech 256 dataset. Caltech 256 is a harder category with more classes and more images than Caltech 101. Due to the variability associated with poses, orientations and some level of occlusion and clutter along with non-class specific data such as background images, Caltech datasets are one of the harder datasets for achieving high classification and detection accuracy. In this chapter a novel method for extracting unique features representable in a high dimensional space is proposed. In addition to this a new method of representing these unique features through sparse representation is discussed along with the use of a good classifier such as AdaBoost. We start with the necessity of the proposed method and introduction followed by detailed analysis and experiments followed by conclusions.

Figure 3.1: Sample Images from Caltech 101 and Caltech 256 dataset.

3.1 Necessity for Image Classification

Images in general are captured under a diverse set of conditions. An image of the same object can be captured with varied poses, illuminations, scales, backgrounds and probably different camera parameters. The task of image classification then lies in forming features of the input images in a representational space where classifiers can be better supported in spite of the above variations. Existing methods have mostly focused on obtaining features which are invariant to scale and translation, and thus they generally suffer from performance degradation on datasets which consist of images with varied poses or camera orientations. Here we present a new framework for image classification, which is built upon a novel

way of feature extraction that generates largely affine-invariant features called affine sparse codes. This is achieved through learning a compact dictionary of features from affine-transformed input images. Analysis and experiments indicate that this novel feature is highly discriminative in addition to being largely affine-invariant. A classifier using AdaBoost is then designed using the affine sparse codes as the input. Extensive experiments with standard databases demonstrate that the proposed approach can obtain the state-of-the-art results, outperforming existing leading approaches in the literature.

3.1.1 Introduction

Image classification has seen significant development in recent years, with new approaches ranging from bag-of-features-based visual vocabulary generation [45] and spatial pyramid matching (SPM) [51] to the most recent locality-constrained linear coding (LLC) [58]. In general, naturally-captured images from various sources are not restricted to fixed acquisition condition. This poses a challenge in terms of associating invariant features to images of the same object under diverse acquisition conditions. Many of the current state-of-the-art image classification framework rely on a set of features which are largely scale and translation invariant. Scale and translation invariant features generally work well for objects with similar poses or in cases where similar features for an object class can be generated by normalizing the pose. But these features may not be discriminative enough when the images involve a wide range of pose variation.

The SPM method [51] formulates the image classification problem in terms of the global non-invariant representation by aggregating local features over different subregions at different scales. This method is effective only when objects involved undergo spatial translation. A non-parametric nearest neighbor classifier [63] obtained good classification performance based on nearest neighbor distances on local image descriptors. But this method is only scale-invariant. Recently

sparse-coding-based SPM method was found to be effective in obtaining promising results on the Caltech datasets [59]. The main idea was the use of sparse codes to obtain discriminative features which could be classified by a classifier such as a linear SVM. The same authors further improved on the performance through the use of LLC, reporting state-of-the-art classification performance on the Caltech 101, the Caltech 256 and the PASCAL datasets [58]. Again the features used in this method were only scale and translation invariant and features would lose their discriminative ability under large pose variations.

Various image categorization datasets such as the Caltech and the Visual Object Class (VOC) datasets have widely varied poses/orientations. This poses a challenging task of obtaining unique features which are discriminative in nature and also largely invariant to common variations including scale, translation and (both in-plane and out-of-plane) rotation. Assuming the commonly-used affine model for image transformation, the problem is then one of finding affine-invariant features. Techniques for image matching using affine transform (e.g., [56]) can be used to generate affine-invariant descriptors. However, such descriptors directly generated from raw image patches are often not discriminative enough on their own. This demands new ways of extracting discriminative features from the raw affine-invariant descriptors. Further, images from multiple classes may have similar appearance, and hence the features, even if being discriminative, may not be sufficient to clearly distinguish the images beyond reasonable doubt.

Aiming at addressing the above challenges, in this thesis we present a new framework for image classification, which is built upon a novel way of feature extraction that generates largely affine-invariant features called affine sparse codes. This is achieved through learning a compact dictionary of features from the set of raw affine-invariant descriptors computed from the input images. Then a classifier using AdaBoost is designed using the affine sparse codes as the input, further im-

proving the separability of the classes by assigning different set of weights to each of the classes adaptively. We evaluated the proposed framework and algorithms based on two commonly-used datasets: Caltech 101 and Caltech 256. Comparative study of the experimental results has shown that the proposed method is able to outperform existing state-of-the-art in the literature.

3.2 Proposed Approach

In this section, we present the proposed approach towards image classification. The proposed method relies on a combination of three key techniques to achieve the desired invariance and accuracy: (1) Exacting affine-invariant raw descriptors from the input images using a simplified Affine-Scale invariant feature transform (ASIFT) algorithm [56]; (2) Developing a novel way of extracting discriminative features through first learning a compact dictionary from the raw descriptors and then perform sparse coding with the dictionary; (3) Building a classifier using AdaBoost to maximally exploit the compact affine sparse codes in final classification. The implementation of the proposed method involves the following logical steps:

1. Obtain ASIFT features for the given input images;
2. Obtain a compact codebook from the dense ASIFT descriptors;
3. Use sparse coding for extracting coefficients from the ASIFT descriptors under the codebook;
4. Select the best descriptor for each spatial region on the basis of minimum error sparse codes;
5. Max pooling of the sparse feature codes across finer subregions;
6. Use a classifier based on AdaBoost for training and testing the affine sparse codes.

We describe the different steps of the algorithm in detail in the following subsections

Figure 3.2: A few examples of Caltech 101 and Caltech 256 dataset showing different poses and orientations in images.

3.2.1 ASIFT: An Overview

SIFT method combines the idea of simulation and normalization [53]. Since scale changes result in blurring of the original image, it cannot be normalized. SIFT obtains invariant features by simulating zoom across different scales. The translation and spin parameters are normalized. In general a camera model involves 6 parameters namely scale, translation (vertical and horizontal), rotation, latitudinal and longitudinal camera axis parameters. Any affine map (without translation) involves transformation through the matrix given by

$$A = \lambda \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} t & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \tag{3.1}$$

where $\lambda > 0$, factor t is responsible for the tilt involved, and ψ represents camera spin and $\phi \in [0,\pi)$. As with SIFT, ASIFT also normalizes translations and spin but it also involves simulation of camera axis parameters and the scale (zoom) parameter.

A smaller dataset like Caltech 101 has large inter-class variations while a much larger dataset like Caltech 256 has large intra-class variations in addition to the inter-class variations. Large intra-class variations along with many similar inter classes put a constraint on how features can be obtained which can be separated in high-dimensional space such that objects belonging to the same class

Figure 3.3: A few examples of Caltech 101 and Caltech 256 dataset showing similar appearance among objects belonging to different classes.

are easily differentiated from other objects of similar classes. A simple example is shown in Fig. 3.2 where an object belonging to the same class has widely varied poses/orientations and scales. These images have different discriminative features and they need to be mapped onto a uniquely representable discriminative feature space. This example illustrates the necessity of a feature transform which is invariant not only to scale but also to varied poses and orientations. While another example in Fig. 3.3 shows objects belonging to different classes which have similar appearances. This makes it extremely difficult to obtain good classification performance on classes with similar features. This example indicates the necessity of a classifier which can discriminate classes with similar features by assigning different weights to them and generating multiple hypothesis.

Images undergo affine distortion caused by change in the optical axis orientation as viewed from a frontal position. These distortions can be charaterized by the latitude and longitude camera parameters ϕ and θ. The longitude parameter also known as ϕ can be simulated by rotating an image about the horizontal axis viewed from the frontal position. The latitude parameter also known as tilt which is inversely related to cosine of the angle θ can be simulated by performing directional t-subsampling defined in [56][60]. The ASIFT framework defined in [56] experimentally provides a set of 6 different tilts performed on a finite number of rotational angles ϕ. Since the image datasets considered comprise of data where

there are no images rotated greater than 90 degrees in the horizontal and vertical axes, we restrict ourselves to a maximum of 4 tilts and corresponding different rotations. So the algorithm in simple terms can be explained as follows:

1. Obtain tilt factor $\mathbf{t} = \sqrt{2}^i$ where i = 1,2,3,4
2. Obtain ϕ for each tilt factor $\mathbf{t}$ given by $\frac{k*72}{\mathbf{t}}$ where k = 1,2,3,... such that $\frac{k*72}{t} < 180°$.
3. Calculate the affine transform of the input image for all tilts $\mathbf{t}$ and rotations ϕ .

The tilts $\mathbf{t} = \frac{1}{\cos\theta}$ correspond to the latitude angle θ and the sampling range follows a geometric series given by $1, a, a^2, \cdots a^n$. Experimentally it has been found that setting a = $\sqrt{2}$ provides a good range for performing various tilts [56]. The longitude angle ϕ for each tilt is sampled accordingly so as to follow an arithmetic series given by $0, \frac{b}{t}, \cdots \frac{kb}{t}$ where b = 72° is a good choice and k is such that $\frac{kb}{t} < 180°$. A set of affine transformed images are obtained using the above method. Dense SIFT descriptors are obtained for each affine transformed image. These dense ASIFT descriptors form the input to the dictionary learning algorithm as well as for the formation of sparse descriptors.

3.2.2 Codebook formation and sparse descriptor generation

The features extracted from ASIFT correspond to a large set of dense descriptors. There exists a lot of redundancy in the descriptors obtained. The most relevant descriptors among them need to be picked. In order to achieve good classification performance we need to generate similar codes for descriptors belonging to the same class and they also should be able to distinguish themselves from descriptors belonging to other classes. Such codes are obtained through sparse representation. This necessitates the need for a prior learned dictionary for which we propose an online learning algorithm. Consider a dictionary D of K basis atoms and dense

features F, then the dense features can be uniquely represented in a dictionary D through sparse representation given by

$$\alpha \cong argmin_{\alpha \in \Re^K} \frac{1}{2} \parallel \mathbf{F} - D\alpha \parallel_2^2 + \lambda \parallel \alpha \parallel_1 \tag{3.2}$$

Under mild conditions the solution to the system is unique. With this background we will consider the codebook formation step.

The ASIFT descriptors obtained are of the order of 10^6. A batch processing based scheme like [64] would require huge amount of memory and also would require lot of computations to obtain accurate representation of the large data features. Thus we resort to an efficient online dictionary learning mechanism. Recently an online dictionary learning scheme of [54][55] details the efficiency of stochastic gradient approximations. For large datasets the speed and memory requirements would be huge and it would be impractical to use a batch processing based optimization technique.

The codebook generation algorithm involves two important steps. The first step is the sparse coding step which involves finding the coefficients which can approximately represent the input features through a dictionary. The second step is dictionary updating which involves updating the base atoms of the dictionary through coordinate descent method with warm restarts. Once the compact dictionary is obtained, the dense ASIFT descriptors can be represented in a dictionary basis through sparse coefficients. The l_1 sparse coding problem can be cast as Eqn. 3.2. This problem also known as basis pursuit or Lasso has been quite successful in l_1-decomposition problems. Since there are two parts in the equation, namely the least squares part and the l_1 penalty part, they can be individually optimized keeping the other one fixed. It is well known that a penalty such as l_1 will lead to a sparse set of coefficients α. We also performed experiments on the sparse coding problem with separable constraints. In this method we write the

Figure 3.4: Plot of error between original and reconstructed features for a few classes

equation with separable positive and negative constraints. It is given by

$$\alpha \cong argmin_{\alpha \in \Re^K} \frac{1}{2} \parallel \mathbf{F} - D_+\alpha_+ + D_-\alpha_- \parallel_2^2 + \lambda\alpha_+^T 1 + \lambda\alpha_-^T 1$$
$$\text{s.t } \alpha_+, \alpha_- \geq 0 \tag{3.3}$$

Eqn. 3.3 is again a convex optimization problem which can be solved using coordinate descent method. Coordinate descent methods are fast and has been shown to converge to a stationary point of the cost function with probability one [48].

Experiments have been conducted on features using the K-nearest neighbours based LLC method, the Lasso method and the coordinate descent method. Fig. 3.4 shows the average squared error over all dimensions of the input features. In the plot, only LLC and Lasso methods have been shown and the errors have been plotted for 30 of the 257 classes of the Caltech 256 dataset. The errors obtained using coordinate descent method (not shown in plot) are comparable with the Lasso method and both of these methods have considerable gain over the K-nearest neighbor based LLC method. One of the reasons why coordinate descent method performs better than others is because of the nature of dictionary updates in the online learning mechanism. Since similar mechanisms are used in both the cases, the codes obtained are much closer to the input features.

The aforementioned algorithm for online dictionary learning is summarized below:

Algorithm 2.1

Online codebook generation for obtaining sparse codes

Input : *Features* $\mathbf{F} \in \Re^{MxN}$, Initial Dictionary

$\mathbf{D}_0 \in \Re^{MxK}$, Iterations $\mathbf{R}, \lambda \in \Re$

(regularization/sparsity parameter)

Output: Dictionary $\mathbf{D} \in \Re^{MxK}$

1 : $\mathbf{P}_0 \leftarrow 0, \mathbf{Q}_0 \leftarrow 0$

2 : for i = 1 to R do

3 : Draw samples $\mathbf{f} \in \Re^M$ from $\mathbf{F}$

Sparse coding step using LASSO

4 : $\alpha_i \cong argmin_{\alpha \in \Re^K} \frac{1}{2} \parallel f_i - D_{i-1}\alpha \parallel_2^2 + \lambda \parallel \alpha \parallel_1$

5 : $P_i = P_{i-1} + \alpha_i \alpha_i^T$

6 : $Q_i = Q_{i-1} + f_i \alpha_i^T$

7 : Calculate D using coordinate descent updates from

$D_{i-1}$and also P_i, Q_i

$D_i \cong argmin_{D \in C} \sum_{j=1}^{i} \frac{1}{i} \parallel f_j - D\alpha_j \parallel_2^2 + \lambda \parallel \alpha_j \parallel_1$

8 : end for

9 : Return D_R

3.2.3 Feature selection via sparse coding

ASIFT descriptors are obtained for various rotations and tilts. Thus we have a multitude of dense feature descriptors for each spatial position of the image. Feature selection involves selecting a subset of features from all the representative

features. We use sparse coding to obtain the best feature for the given spatial location among all ASIFT descriptors. Let A_k be the descriptor for the k^{th} affine transformed image and let f_k be the descriptor obtained by sparse representation of A_k, we select the best descriptor given by

$$\mathbf{A}_k \cong \min_{k=1}^{L} \parallel A_k - f_k \parallel_2^2 \tag{3.4}$$

where L is the number of affine transformed images on which SIFT descriptors are formed. Thus among all the dense ASIFT descriptors for each spatial location, only one of the sparse code gets selected. The assumption is that the low error sparse codes are more likely to lead to informative and discriminative codes than the ones with higher error. There are two advantages of picking the code with the lowest error. First, the codes are the best representations of the input feature; Second, when the error is small, the codes are sparser, resulting in larger coefficient values. Larger coefficients inherently lead to selection of the closest basis from the dictionary for the input feature during max-pooling. This method thus plays an important role in spatial pooling where sparse codes are max-pooled. Spatial max-pooling involves dividing the image into finer sub-regions and picking the largest coefficient among the sparse coefficients obtained from the ASIFT dictionary. The largest coefficient represents the weightage associated with the dictionary element and uniquely represent the feature for the spatial region. Codes formed across different sub-regions are now concatenated to obtain the final feature descriptors. These feature descriptors form input to the classifier.

3.2.4 AdaBoost-based Classification

Feature extraction, representation and selection are necessary for formation of the training and test sets for a classification algorithm. An efficient classifier would make the best usage of the given training data set to learn the model and gener-

alize it over the test data. Recognizing that boosting is one such general method for improving the accuracy of any given learning algorithm [46][47], in this work, we propose to use AdaBoost [62] in building the desired classifier. For the multi-class case, the AdaBoost algorithm takes input features for all different classes with different labels. It calls a weak learning algorithm repeatedly for a different distribution set over different classes. The distribution for all classes represents the weights associated with each sample belonging to each class. Initially the distribution is uniform, and after each iteration the weak classifier returns a hypothesis. The distribution is modified so as to give more weightage to misclassified samples of each class. The error of the weak learner's hypothesis is measured by its misclassified samples on the distribution on which the samples were trained. The weak hypothesis outputs the classification accuracy based on the distribution of the samples. In case of binary class, even if the error is greater than $\frac{1}{2}$ the hypothesis '$h(x_i)$'can be replaced by '$1 - h(x_i)$'[46]. Hence theoretically we can minimize the classification error as small as possible until overfitting occurs. However, in the multi-class case this cannot be done because there cannot be an equivalent of hypothesis '$1 - h(x_i)$'in the multiclass case and hence we need to stop continuing with generating the hypothesis once classification accuracy is less than $\frac{1}{2}$.

With these, we summarize the actual implementation of the AdaBoost algorithm used in this thesis:

Algorithm 2.2

Implementation of Multiclass AdaBoost Algorithm of [46]

Input : Sequence of training and testing features

ftrain, ftest $\in F$ with labels $\mathbf{y}_i \in Y$

1 : Initialize weights $D_1, D_2, \cdots D_N = \frac{1}{N}$

2 : for j = 1,2,...T

3 : Call weaklearning algorithm such as SVM with

distribution D; get back the model and hypothesis h_j

4 : Error over $D : \epsilon_j = \sum_{i=1}^{N} D_i^j [h_j(x_i) \neq y_i]$

5 : If $\epsilon_j > \frac{1}{2}$terminate loop

6 : Using model obtain testing hypothesis H_j

7 : Calculate $\beta_j = \frac{\epsilon_j}{1-\epsilon_j}$

8 : Calculate weights $D_i^{j+1} = D_i^j \beta_j^{1-[h_j(x_i) \neq y_i]}$

9 : end for

10 : Output final train hypothesis

$\mathbf{h}_T(F) = argmax_{\mathbf{y} \in \Re^Y} \sum_{j=1}^{T} \log(\frac{1}{\beta_j})[h_j(F) = y]$

11 : Output final test hypothesis

$\mathbf{H}_T(F) = argmax_{\mathbf{y} \in \Re^Y} \sum_{j=1}^{T} \log(\frac{1}{\beta_j})[H_j(F) = y]$

3.3 Experimental Results

The experiments were performed on the Caltech 101 and Caltech 256 datasets. We used only ASIFT descriptor for all the experiments. The dimension of each ASIFT descriptor is 128. The set of descriptors of the order of 10^6 are trained using the online dictionary learning mechanism to obtain a dictionary of size 1024.

ASIFT descriptors generated from images taken only from Caltech 256 dataset were used for training a common dictionary which was used for sparse descriptor generation for both Caltech 101 and Caltech 256 dataset. The best affine sparse descriptors obtained after feature selection are max-pooled across 4x4,2x2 and 1x1 scales to obtain the final feature descriptors. The max pooling is obtained by selecting the max of the sparse codes obtained across different sub regions. These codes are now concatenated to obtain a final feature vector which is sparse.

3.3.1 Results with Caltech 101

Table 3.1 shows the results obtained for the Caltech 101 dataset. Caltech 101 dataset consists of 9144 images which are divided among 101 object classes and 1 background class. As we can see from Table 3.1, even for a small training size the classification accuracy is comparatively higher than other methods. The classification performance without the background class for train size of 30 is 87.72%. The percentage accuracy for various classes is illustrated in Fig.3.5 and Fig.3.6. As we can see from Fig.3.5, a few of the classes achieved 100% accuracy. In fact a total of 8 classes achieved 100% accuracy. We also provide a few examples with accuracy less than 25%, shown in Fig. 3.6. As expected, the background class is one among them since there are no specific features which are discriminative and hence leading to misclassification. The other cases includes cougar body which was in majority classified as leopard, and crab as crayfish. These are typical examples of classes which are extremely similar in nature and are hard to classify even with the most discriminative features. Other factors include the camouflaging of images with the background and occlusion. Over 70 classes achieve an accuracy of 50% or higher. Only 5 classes had low accuracy of 25% or less.

Experiments on classification performance with and without AdaBoost was also carried out. This is illustrated through the use of another classifier such as SVM. Table 3.2 illustrates the performance of a classifier such as SVM with

Gerenuk	Accordion	Skate	Sunflower	Umbrella
Acc-100	Acc-100	Acc-100	Acc-98.8	Acc-98.3

Figure 3.5: Results of Caltech 101 dataset showing some selected classes with high accuracy.

Background	Beaver	Mayfly	crocodile	Crab
Acc-13	Acc-18.9	Acc-22	Acc-23	Acc-23.5

Figure 3.6: Results of Caltech 101 dataset showing some selected classes with low accuracy.

Table 3.1: Caltech 101 dataset classification results

Training size	5	10	15	20	25	30
Zhang[61]	46.6	55.8	59.1	62	-	66.2
Lazebnik[51]	-	-	56.4	-	-	64.6
Griffin[49]	44.2	54.5	59.0	63.3	65.8	67.6
Boiman[63]	-	-	65.0	-	-	70.4
Jain[50]	-	-	61.0	-	-	69.1
Gemert[57]	-	-	-	-	-	64.16
Yang[58]	51.15	59.77	65.43	67.74	70.16	73.44
Ours	66.13	73.09	78.38	78.50	82.36	83.28

Note: '-' indicates unavailability of results

linear kernel on Caltech 101. Similarly, Table 3.4 illustrates the same for the Caltech 256 dataset. Although [52] emphasizes on the effectiveness of radial basis functions as a kernel, we used a linear SVM kernel because of low computational complexity involved in training. Using an SVM with linear kernel as a weak learner, we obtained a classification accuracy of 79%. The training involved in case

Table 3.2: Performance of SVM and AdaBoost on the Caltech-101 dataset

Training size	5	10	15	20	25	30
SVM	63.4	70.1	73.6	73.9	77.3	78.9
AdaBoost	66.13	73.09	78.38	78.50	82.36	83.2

of AdaBoost was not intensive. Only three iterations were required to train the weak classifier and obtain a hypothesis for each case. It is obvious that, without involving intense training, there has been considerable performance gain achieved by AdaBoost. The classes for which the performance was improved in each of the hypothesis were the ones whose images were largely similar. The distribution change was able to convert the misclassified samples to their respective class without affecting the appropriately classified samples. We shall see how error bounds affect the classification performance of AdaBoost in a later sub-section.

3.3.2 Results with Caltech 256

Table 3.3 shows the results for Caltech 256. This is a harder dataset with much larger inter as well as intra class variations. There are a total of 30607 images which are divided among 256 object classes and 1 background class. Fig. 3.7 provides accuracies for a few of the classes in Caltech 256. The dictionary used in the sparse descriptor generation consists entirely of images only from Caltech 256 dataset. Experiments were carried out on online dictionary training using 40%, 80% and 100% of the images from Caltech 256 dataset. A common dictionary trained from such images was used for feature descriptor generation in both Caltech 101 and 256 datasets. There was no significant difference in the performance obtained when the number of images used were reduced from 100% to 80% and to 40% for Caltech 256 dataset. In fact, in case of Caltech 101 there was slight increase in the performance when 80% and 40% images were used, which may be because of overfitting issues when more number of images are involved. Table 3.5 shows some of the results obtained for Caltech 256 and Caltech 101 datasets

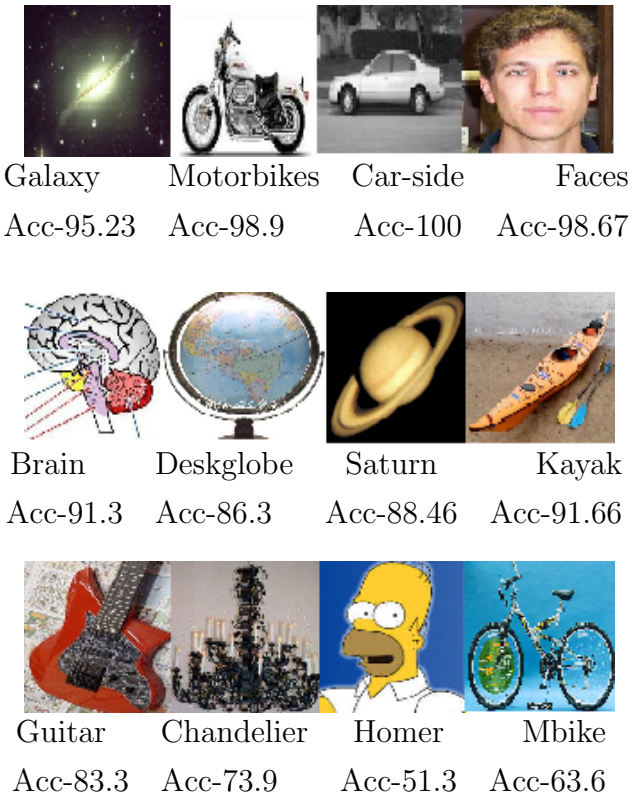

Figure 3.7: Results of Caltech 256 dataset showing classes with different accuracies.

when different percentage of the images were selected for dictionary learning. For Caltech 256 dataset in cases when 80% and 40% of images were used in dictionary learning, it was made sure that the remaining 20% and 60% images would be part of the test set. For the Caltech 101 case no such restrictions were involved for training and testing. This is a clear indicator that a single dictionary generated from a larger dataset would result in discriminative codes for both Caltech 101 and Caltech 256. This again substantiates the discriminative power of the dictionary for generating sparse codes which are largely affine-invariant.

3.3.3 Analysis of affine sparse codes

The affine sparse descriptors are discriminative in nature. The reason is attributed to the sparse coefficients obtained which can be termed as features with minimum intra class variance and maximum inter class variance. This comparison was

Table 3.3: Image classification results for Caltech-256 dataset

Training size	15	30	45	60
Griffin[49]	28.3	34.1	-	-
Gemert[57]	-	27.17	-	-
Yang[58]	34.36	41.19	45.31	47.68
Ours	39.42	45.83	49.3	51.36

Table 3.4: Performance of SVM and AdaBoost on Caltech-256 dataset

Training size	15	30	45	60
SVM	37.67	43.1	46.9	49.84
AdaBoost	39.42	45.83	49.3	51.36

Table 3.5: Performance comparison on images selected for dictionary learning

Training size	15	30	60
Caltech256(40%)	37.3	44.11	49.2
Caltech256(80%)	38.51	45.24	51.13
Caltech256(100%)	39.42	45.8	51.36
Caltech101(40%)	79.98	84.1	-
Caltech101(80%)	79.2	83.8	-
Caltech101(100%)	78.38	83.2	-

made with the SIFT LLC codes. Correlation statistics for affine sparse codes are shown in Fig.3.9 and SIFT codes are shown in Fig.3.8. Fig.3.10 shows the sum of correlations obtained for each class. The intra-class correlations obtained for the same class of features represent within class correlations among feature vectors. The inter class correlations represent the correlations between feature vectors belonging to different classes. A random set of feature vectors were correlated with a random set of vectors from all other classes. The number of random vectors picked for each class was 30. The number of random classes picked to correlate with the current class was 25. The four different colors shown in Fig. 3.10 shows four different correlation statistic of the two different codes. As can be seen from

Fig. 3.10, the red and green labels clearly indicate that affine sparse codes have higher intra class correlations and lower inter class correlations than SIFT LLC codes shown in blue and black labels respectively. This is also evident from the scatter matrix plots of fig 3.8 and fig 3.9. The scatter matrix is a representation of the pearson correlation coefficient statistic. The points represent the scatter of each class with respect to every other class.

Correlations are divided into three different ranges as can be seen in Fig. 3.8 and Fig. 3.9. High correlation values, mid and low correlation values are represented by black dots, red dots and green dots respectively. Black dots clearly seen on the diagonal indicate the correlation among class features of the same class. Red and green dots indicate correlations of each class feature with features of other classes. Both sparse codes and LLC codes exhibit higher correlations among features of same class. But sparse codes gain an upper hand in terms of inter-class correlations. We can see denser red dots in case of LLC codes indicating higher inter-class correlations than in case of affine sparse codes. Sparser red dots lead to lower inter-class correlations and hence the features are discriminative with respect to each other. Dense green dots obviously imply sparse red dots and hence lower inter-class correlations. Thus the classification performance is improved by the high intra class correlation and low inter class correlation between features. This is quite evident from Table 3.1 and Table 3.3 for both Caltech 101 and Caltech 256 datasets.

3.3.4 Analysis of error bounds of AdaBoost

Suppose that the weak learning algorithm such as SVM generates errors ϵ_1, $\epsilon_2 ... \epsilon_T$ where ϵ_j is defined as shown in **Algorithm 2.2** and assuming $\epsilon_j \leq 1/2$, then error

$$\epsilon = \sum_{\text{i } \tilde{}\text{D}} [h_f(x_i) \neq y_i] \tag{3.5}$$

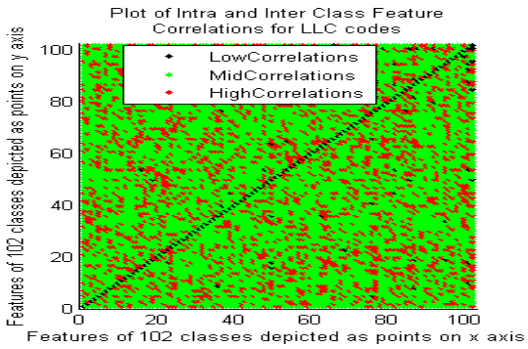

Figure 3.8: Plot of scatter matrix of all classes for LLC codes belonging to Caltech 101 dataset

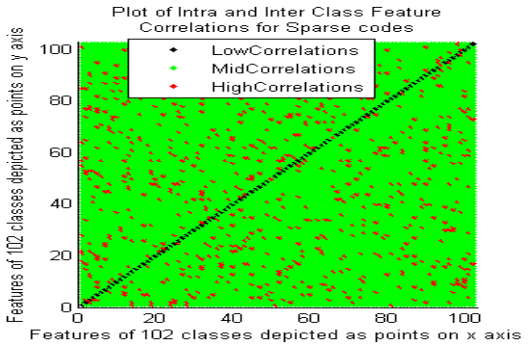

Figure 3.9: Plot of scatter matrix of all classes for Sparse codes belonging to Caltech 101 dataset

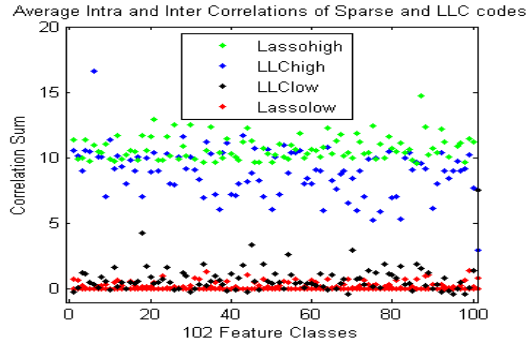

Figure 3.10: Plot of averaged correlations for LLC and Sparse codes

Table 3.6: Error bounds of AdaBoost algorithm on Caltech-101 and Caltech-256 datasets

Dataset	ϵ_1	ϵ_2	ϵ_3	ϵ_f
Caltech101	0.12	0.024	0.0024	$1.4.10^{-3}$
Caltech256	0.263	0.081	0.0259	$3.13.10^{-3}$

defined in [46] of the final hypothesis h_f is bounded by

$$\epsilon \leq 2^T \Pi_{j=1}^{T} \sqrt{\epsilon_j (1 - \epsilon_j)} \tag{3.6}$$

We obtained error bounds for the Caltech 101 and Caltech 256 datasets as shown in Table 3.6. These error bounds also illustrate the fact that beyond a certain number of iterations the error of the final hypothesis would not accurately represent the training error because it would be less than $\frac{1}{2}$ and that would be the point to stop generating hypothesis.

3.4 Conclusion and Discussion

We proposed the affine sparse codes for providing compact and discriminative features, which is then used in an AdaBoost-based classifier for the image classification task. Detailed analysis has been performed on the proposed approach, using two standard test sets. The discriminative nature of the proposed feature is due to the affine-invariance and sparsity-based learning. Sparsity allows us to pick different number of basis atoms from the dictionary and hence leading to low-error high-energy codes. Affine invariance is responsible for low intra-class variance, thus making features of the same class clustered tightly around its mean. With the proposed method, we have seen considerable gain in classification performance over leading existing methods.

One of the drawbacks of the current method is the use of large number of raw descriptors. A new method for efficiently discarding the dense feature points before online dictionary learning needs to be incorporated. This will considerably

reduce the amount of space required to extract each dense descriptor and storing it before sparse coding. Also the existing method may not achieve good performance on datasets involving multiple class labels in a single image. Thus features extracted from the images should be such that multiple labels can be assigned to it by a classifier. Thus we aim at addressing following issues in the future: Obtaining considerably less number of ASIFT descriptors to reduce space requirements and also better feature selection mechanisms to generate unique sparse features of low dimensionality. Combining these with a classifier which has the ability to assign multiple class labels to certain features would lead to much better classification system.

Chapter 4

EXPLORING K-SVD BASED IMAGE DE-NOISING USING MATRIX COMPLETION

In many practical problems of interest, one would like to recover a matrix from a sampling of its entries. In computer vision and image processing, many problems can be formulated as the missing value estimation problem, e.g., image in-painting [66][67][68], video decoding, and video in-painting. The values can be missing due to problems in the acquisition process, or because the user manually identified unwanted outliers. Image denosing has been an active research topic for many many years. Since image noise is generally caused by image sensors, amplifiers, ADC's, or maybe even due to quantization, it is imperative that the noise should be handled by an image denoising algorithm. Image denoising problem in general can be modeled as one of a clean image being contaminated by additive white Gaussian noise (AWGN), though modeling in terms of impulse or Poisson noise is also common. In this thesis we introduce a new method for exploring K-SVD based image denoising through low-rank matrix completion. This method incorporates dictionary formation and learning through sparse representation using K-SVD. Before getting into the details of the new method an overview of Matrix Completion is given.

4.1 Introduction to Matrix Completion and Related Work

Recently, Candes and Recht [69][70] showed that if a certain restricted isometry property holds for the linear transformation of the constraints, the minimum rank solution can be recovered by solving a convex optimization problem, namely the minimization of the trace norm. Their work theoretically justified the validity of the trace norm to approximate the rank [73]. Indeed, they proved that most low-rank matrices can be recovered exactly from most sets of sampled entries even though these sets have surprisingly small cardinality, and more importantly, they

proved that this can be done by solving a simple convex optimization problem. To state their results, suppose that the unknown matrix M $\in \Re^{nxn}$ is square, and that one has available m sampled entries M_{ij} : (i; j) $\in \Sigma$ where Σ is a random subset of cardinality m. [69] proves that most matrices $\mathbf{M}$ of rank r can be perfectly recovered by solving the optimization problem

$$minimize \parallel X \parallel_* \ \ subject\ to\ X_{ij} = M_{ij}, (i,j) \in \Sigma \tag{4.1}$$

provided the number of samples obeys

$$m \geq Cn^{6/5} r log(n) \tag{4.2}$$

for some positive constant C. In the next subsection, an overview of the algorithm and the detailed experimentation are explained.

4.2 Overview of the Algorithm

In this study, the K-SVD algorithm is used in exploring the impact of matrix completion on image de-noising. Our study is based on the premise that an underlying structure exists in the noisy image which can be carried over into a representational space where noisy pixels can be removed to obtain denoised patches which are very close to the original. The algorithm assumes a partially denoised image obtained from the K-SVD algorithm. Then the patches of the denoised image are used in subsequent steps to obtain better patches in the reconstructed denoised image. The following steps outline the algorithm: (i) Obtain a partially denoised image using any of the algorithms such as K-SVD based denoising. (ii) Obtain randomly sampled patches from this partially denoised image across different scales to form different dictionaries. (iii) Train the dictionaries to obtain a better compact representation for these randomly sampled dictionaries. (iv) Collect randomly sampled patches from the noisy image and form a randomly sampled dictionary; Train it using online dictionary learning algorithm to obtain a compact trained dictionary. The only difference is that this is done across one

scale only. (v) Obtain the sparse representation for a noisy patch and use the sparse coefficients to form a patch from all dictionaries generated from partially denoised image. (vi) Use all the patches obtained from different dictionaries to form a matrix. Remove pixels which are noisy through the means of comparing the variances of partially denoised patch and sparse representation based patches. In addition to this, thresholds are also determined using pixel difference between K-SVD denoised patches and noisy patches. (vii) Subject this matrix with missing entries to matrix completion. The recovered matrix represents the completely denoised patch. This process is repeated for all patches of an image.

4.2.1 Dictionary formation and learning

This is the first step of the algorithm. Once a partially denoised image is obtained through K-SVD, this image is used for randomly sampling overlapping patches to obtain a randomly sampled dictionary. For the experimentation, five different sets of randomly sampled dictionary were used. In addition to these three scales were used for forming these dictionaries. So in addition to the original scale two downsampled scales were used to obtain randomly sampled patches. Thus we have a total of fifteen randomly sampled dictionaries across three scales. Now these dictionaries are trained using an online dictionary learning algorithm to obtain a compact learned dictionary. These dictionaries are further used for representing patches obtained from the noisy image. In addition to these fifteen dictionaries, a noisy dictionary of randomly sampled patches is formed. This dictionary is trained to obtain a noisy trained dictionary.

4.2.2 Sparse representation and noise removal

The next step is sparse representation. Given a noisy patch, a sparse representation of this noisy patch from the noisy dictionary is formed. These coefficients are carried over to form an image patch from all the fifteen dictionaries. Now these representation individually may represent a recovered image itself. But these may

not be the best denoised image that can be formed since each dictionary can at best represent the original partially denoised patch itself. Hence an appropriate method of noise removal is to be undertaken. Based on the variance of the image patches a different threshold is set to determine pixel values which are far away from partially denoised image. The noisy image is used to provide an input on the variance of the patch and the variability of individual pixels to aid the pixel removal step. Now these patches with noisy pixels removed are arranged to form a large matrix.

4.2.3 Matrix completion of sparse representation based patches

Now the large matrix with missing entries obtained from sparse representation is subjected to matrix completion. Matrix completion is a method of recovering missing entries of a sufficiently low-rank matrix through nuclear norm minimization. Mathematically this can be represented as a matrix with missing entries $M_{j,k}$. The matrix recovery involves solving the minimization problem from the incomplete set of observations $M_{j,k}$ to obtain $N_{j,k}$ given by

$$min_N \parallel N \parallel_* s.t. \parallel N|_\Omega - M|_\Omega \parallel_F^2 \leq \#(\Omega)\hat{\sigma}^2 \tag{4.3}$$

where $\hat{\sigma}$ is the estimate of standard deviation of the noise, which is obtained by calculating the average of the variances of all elements $\in \Omega$ on each row where Ω is the index set where $M|_\Omega$ denotes the vector including elements in Ω only. Instead of solving 4.3 directly a lagrangian version is solved which is given by

$$min_N \ 0.5* \parallel N|_\Omega - M|_\Omega \parallel_F^2 + \mu \parallel M \parallel_* \tag{4.4}$$

which is equivalent to 4.3 for some value of μ by the duality theory. There are many efficient algorithms available for solving the minimization problem of 4.4. The fixed point iterative algorithm is used in this implementation and the detailed algorithm is as shown below in Algorithm 1.

Algorithm 1

Fixed point iterative algorithm for solving the minimization problem of 4.4

$$1.Set\ N^{(0)} := 0$$

$$2.Iterating\ on\ i\ till \parallel N^{(i)} - N^{(i-1)} \parallel_F \leq \epsilon$$

$$\begin{cases} Z^{(i)} = N^{(i)} - \tau M_\Omega(N^{(i)} - M) \\ N^{(i+1)} = D_{\tau\mu}(Z^{(i)}), \end{cases}$$

where τ and $1 \leq \tau \leq 2$ are pre-defined parameters, D is the shrinkage operator defined as $D_\tau(M) = U\Sigma_\tau V^T$ and N_Ω is the projection operator of Ω defined by

$$M_\Omega(i,j) = \begin{cases} N(i,j),\ if(i,j) \in \Omega \\ 0,\ otherwise. \end{cases}$$

$$3.Output\ N := N^{(i)}$$

4.3 Experiments and Visual Results

In our experiments fifteen patches reconstructed from the sparse representation are chosen to obtain fifteen vectors which are stacked to form the large matrix. The variance of the reconstructed patch was used as the threshold. In addition to this, the pixel difference between the denoised K-SVD image and noisy image was also used as an additional constraint. The threshold is used to compare the pixel difference between the denoised K-SVD image and the dictionary based reconstructed image. Based on this threshold the pixels are removed from the reconstructed image. For matrix completion, the stopping criterion used is either one of $\epsilon \leq 10^{-5}$ or the maximum number of iterations 500 being reached, whichever occurs first.

The final results are compared with the original to understand the measure of accuracy obtained from many missing entries. Fig.4.1 shows an original

Figure 4.1: Original Image

Figure 4.2: Image corrupted with Gaussian Noise

image to which is corrupted with a gaussian noise as shown in Fig.4.2. There are two reconstructed images shown here with Fig.4.3 image denoised using the K-SVD method of [71] and Fig.4.4 image denoised using matrix completion. The mean square error obtained is slightly higher than the one obtained using K-SVD. With better approach towards removing noisy pixels, there is a better chance of recovering a denoised image very close to the original.

The table 4.1 illustrates some of the results obtained on different number of patches of different images. Approximately 40% of the patches obtained using matrix completion have lower mean squared error than de-noised K-SVD patches. The table shows statistics of the number of patches obtained using matrix completion which have better mean squared error than de-noised with K-SVD for

Figure 4.3: Image denoised using K-SVD method

Figure 4.4: Image denoised using Matrix Completion method

Table 4.1: Statistics of Patches reconstructed using K-SVD and Matrix Completion

Image	Total Patches	No of patches with better MSE obtained using Matrix Completion	No of patches with better MSE obtained using K-SVD
Boat	3969	1313	2656
Bridge	3969	1426	2543
Couple	3969	1361	2608
Man	3969	1541	2428

different images and vice-versa. Original groundtruth patches were used to compare the mean squared error for de-noised K-SVD and matrix completion patches. This empirically proves that with better noisy pixel removal techniques, better than de-noised K-SVD method can be obtained. In addition to this, a prior knowledge on the texture of the patches would aid in picking the appropriate patches for combined reconstruction using K-SVD and matrix completion, eliminating the need for groundtruth patches.

4.4 Conclusions and Future Work

K-SVD based de-noising algorithm is explored through matrix completion. A good percentage of patches can be reconstructed which are very close to the original and have lower mean squared error than those obtained using K-SVD. Under the assumption that a noisy image has an underlying structure which is able to be represented in an already existing denoised image, we can formulate the problem of forming similar patches as a sparse representation problem. Once all the sparse representation based patches are obtained there are stacked to formulate the denoising problem as a matrix completion problem.Prior to applying matrix completion the noisy pixels are removed to obtain missing entries in a largely stacked patch matrix. This method does not assume any underlying statistical properties of image noise and is robust to patch matching error. The advantage of this method is the use of single image only for denoising eliminating the need for storing many images which generally is the case with denoising.This method is also robust different types of noise since no noise property is used for denoising purposes. Future work involves finding the appropriate textured patches to eliminate the use of groundtruth image patches.There is also a need to explore single image denoising using as few dictionaries as possible. A thorough analysis of pixel removal to appropriately remove the noisy pixels only need to be examined, which in combination with finding appropriate textured patches might provide a basis

for single image de-noising using matrix completion only.

Chapter 5

CONCLUSIONS

In this thesis, three pieces of closely-related studies were reported. First, a new framework for understanding and analyzing CS based SR is proposed. The simulation results and analysis clearly show that sparse recovery and representation are different aspects of the problem in CS and hence similar properties of CS may not hold true in sparse recovery case. Visual results which provided consistent results among trained dictionaries further support the argument that trained dictionaries are better than randomly sampled dictionaries. This thesis also proposes a new framework for image classification. A new way of representing images in an unique subspace through affine projection is proposed. A dictionary learning algorithm based on online-learning is developed. The affine sparse codes are generated through the dictionary and classified through one of the boosting algorithms namely AdaBoost. Results on the standard databases affirm that the codes are indeed unique and can result in state of the art results on publicly available datasets. Finally, a new method for obtaining high quality image patches over existing denoising algorithms is proposed and implemented. Sparse representation and matrix completion techniques are utilized on the image to be denoised to obtain high quality denoised image patches. Results confirm the existence of substructure within noisy image which can be extracted to obtain high quality image patches. Though the results are not consistent across all patches of the image, these results provide impetus for selecting appropriate thresholds for different textured patches to obtain consistency across all patches.

REFERENCES

[1] J Sun, ZB Xu, HY Shum. Image super-resolution using gradient profile prior. CVPR 2008.

[2] S.Y. Dai, M. Han,W. Xu, Y.Wu, and Y.H. Gong. Soft edge smoothness prior for alpha channel super resolution. CVPR 2007.

[3] H. A. Aly and E. Dubois. Image up-sampling using total-variation regularization with a new observation model. IEEE Trans. on IP, 14(10):16471659, 2005.

[4] R. Schultz and R. Stevenson, Extraction of high-resolution frames from video sequences. IEEE Trans. on Image Processing, 5(6):996 1011, 1996.

[5] M.S. Lewicki and T.J. Sejnowski, Learning overcomplete representations. Neural Computation, 12(2):337365, 2000.

[6] M. Irani and S. Peleg. Motion analysis for image enhancement: resolution, occlusion and transparency.JVCI 1993.

[7] J. Yang, J. Wright, T. Huang, Y. Ma, Image Super-Resolution as Sparse Representation of Raw Image Patches, CVPR 2008.

[8] H. Rauhut, K. Schnass, P.Vandergheynst, Compressed sensing and redundant dictionaries, IEEE Trans. on Information Theory, Vol. 54(5), May 2008, p 2210-19.

[9] M. Elad, Optimized projections for compressed sensing. IEEE Trans. on Sig. Proc., v 55, n 12, Dec. 2007, p 5695-702.

[10] M. Aharon, M. Elad and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. on Signal Processing, 54(11):43114322, November 2006.

[11] M. Aharon and M. Elad. Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. on Image Proc, 15(12):37363745, December 2006.

[12] H. Chang, D.-Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. CVPR 2004.

[13] R. C. Hardie, K. J. Barnard, and E. Armstrong, Joint map registration and high-resolution image estimation using a sequence of undersampled images. IEEE Trans. on Image Processing, 6(12):16211633, 1997.

[14] R.A. DeVore, Deterministic constructions of compressed sensing matrices Journal of Complexity, v. 23, p. 918-25, 2007.

[15] E. Cands and J. Romberg, Practical signal recovery from random projections. Wavelet Applications in Signal and Image Processing XI, Proc. SPIE Conf. 5914.

[16] E. Cands, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Information Theory, 52:489509, 2006.

[17] David L., Donoho, J.TannerCounting faces of randomly-projected polytopes when the projection radically lowers dimension; Journal of the AMS, Vol. 22(1) (2009) 1-53

[18] E. Cand'es, J. Romberg, and T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Comm. on Pure and Applied Math, vol. 59, no. 8, 2006, pp. 12071223.

[19] J.-J. Fuchs, On sparse representations in arbitrary redundant bases, IEEE Trans. on Information Theory, Volume 50, Issue 6, June 2004 Page(s): 1341 1344.

[20] G. Yu, Stphane Mallat, Sparse Super-Resolution with space matching pursuit, SPARS 2009.

[21] H. Lee, A. Battle , R. Raina , A.Y. Ng, Efficient sparse coding algorithms, NIPS, 2007

[22] Baker,S. and Kanade,T "Limits on super-resolution and how to break them" IEEE PAMI 24(9):1167-1183, 2002

[23] J. Sun, N. N. Zheng, H. Tao, and H. Y. Shum. Generic image hallucination with primal sketch prior. In CVPR, 2003

[24] Z. Lin, J. He, X. Tang, and C.-K. Tang. Limits of learning-based superresolution algorithms. In ICCV, 2007

[25] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In ICCV, 2009

[26] Q. Shan, Z. Li, J. Jia, and C.-K. Tang. Fast image/video upsampling. ACM Trans. Graphics, 27(5), 2008

[27] J D van Ouwerkerk "Image super-resolution survey" Image and Vision Computing, 24(10):1039-1052, 2006

[28] Park, S.C. and Park, M.K. and Kang, M.G. "Super-Resolution Image Reconstruction", IEEE signal processing magazine, 2003

[29] D. L. Donoho, Compressed sensing, IEEE Trans. Inform. Theory, vol. 52,July 2006, pp. 12891306.

[30] D.S. Taubman and M.W. Marcellin, JPEG 2000: Image Compression Fundamentals, Standards and Practice. Norwell, MA: Kluwer, 2001.

[31] D.L. Donoho and X. Huo, Uncertainty principles and ideal atomic decomposition, IEEE Trans. Inform. Theory, vol. 47, no. 7, pp. 28452862, Nov. 2001.

[32] E. Cands and T. Tao, Decoding by linear programming, IEEE Trans. Inform. Theory, vol. 51, no. 12, pp. 4203-4215, Dec. 2005.

[33] J.-J. Fuchs, On sparse representations in arbitrary redundant bases, IEEE Trans. on Information Theory, Volume 50, Issue 6, June 2004 Page(s): 1341 1344.

[34] M. Elad and A. Feuer. Restoration of single super-resolution image from several blurred,noisy and down-sampled measured images. IEEE Transactions on Image Processing,6(12):164658, 1997.

[35] D. Capel. ImageMosaicing and Super-Resolution. Springer-Verlag, 2004

[36] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar. Fast and robust multiframe super resolution. T-IP, (10), 2004

[37] M. Irani and S. Peleg. Improving resolution by image registration. CVGIP, (3), 1991.

[38] Gowda R.Perceptual image/video enhancement for digital TV applications.Masters Thesis, Arizona State University, 2009

[38] Fei-Fei, L. and Fergus, R. and Perona, P. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. IEEE. CVPR 2004

[39] Mutch, J. and Lowe, D.G.Multiclass object recognition with sparse, localized features,1063-6919,IEEE Computer Society ,2006.

[40] R Fergus, Visual Object Category Recognition, PhD thesis, University of Oxford,2005

[41] Belongie, S. and Malik, J. and Puzicha, J.IEEE Transactions on Pattern Analysis and Machine Intelligence, 509–522, Published by the IEEE Computer Society,2002.

[42] Nene, S.A. and Nayar, S.K. and Murase, H.Columbia object image library (coil-100), Techn. Rep. No. CUCS-006-96, dept. Comp. Science, Columbia University, 1996.

[43] Torralba, A. and Murphy, K.P. and Freeman, W.T.Sharing features: efficient boosting procedures for multiclass object detection, IEEE Computer Society,2004.

[44] Everingham, M. and Van Gool, L. and Williams, C.K.I. and Winn, J. and Zisserman, A. The PASCAL visual object classes (VOC) challenge, 303–338, vol88, International Journal of Computer Vision.2010

[45] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, page 22. Citeseer, 2004.

[46] Y. Freund and R. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer, 1995.

[47] Y. Freund, R. Schapire, and N. Abe. A short introduction to boosting. *JOURNAL-JAPANESE SOCIETY FOR ARTIFICIAL INTELLIGENCE*, 14:771–780, 1999.

[48] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[49] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[50] P. Jain, B. Kulis, and K. Grauman. Fast image search for learned metrics. 2008.

[51] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

[52] X. Li, L. Wang, and E. Sung. A study of AdaBoost with SVM based weak learners. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, volume 1, pages 196–201. IEEE, 2005.

[53] D. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[54] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 689–696. ACM, 2009.

[55] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, 11:19–60, 2010.

[56] J. Morel and G. Yu. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009.

[57] J. van Gemert, J. Geusebroek, C. Veenman, and A. Smeulders. Kernel codebooks for scene categorization. *Computer Vision–ECCV 2008*, pages 696–709, 2008.

[58] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367. IEEE, 2010.

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

[60] G. Yu and J. Morel. A fully affine invariant image comparison method. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 1597–1600. IEEE, 2009.

[61] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2126–2136. IEEE, 2006.

[62] J. Zhu, S. Rosset, H. Zou, and T. Hastie. Multi-class adaboost. *Ann Arbor*, 1001:48109, 2006.

[63] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[64] H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. *Advances in neural information processing systems*, 19:801, 2007.

[65] Ponce, J. and Berg, T. and Everingham, M. and Forsyth, D. and Hebert, M. and Lazebnik, S. and Marszalek, M. and Schmid, C. and Russell, B. and Torralba, A. and others, Dataset issues in object recognition, pages 29–48, year 2006,Toward category-level object recognition, publisher Springer

[66] N. Komodakis and G. Tziritas. Image completion using global optimization. CVPR, pages 417424, 2006.

[67] T. Korah and C. Rasmussen. Spatiotemporal inpainting for recovering texture maps of occluded building facades. IEEE Transactions on Image Processing, 16:22622271, 2007.

[68] Mairal, J. and Bach, F. and Ponce, J. and Sapiro, G. and Zisserman, A.Discriminative learned dictionaries for local image analysis,2008,IEEE

[69] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. 2008.

[70] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum rank solutions of linear matrix equations via nuclear norm minimization. SIAM.

[71] Adler, A. and Hel-Or, Y. and Elad, M. A weighted discriminative approach for image denoising with overcomplete representations,IEEE International Con-

ference on Acoustics Speech and Signal Processing (ICASSP), pages=782–785,IEEE, 2010

[72] Candes, E.J. and Wakin, M.B. An introduction to compressive sampling, IEEE Signal Processing Magazine, vol 25,pages=21-30, 2008

[73] Liu, J. and Musialski, P. and Wonka, P. and Ye, J.Tensor completion for estimating missing values in visual data, IEEE 12th International Conference on Computer Vision, pages=2114–2121

[74] Szeliski, R. Computer Vision: Algorithms and Applications, Springer-Verlag New York Inc, 2010