Improvements to sparse signal processing in compressive sensing and other methods

Huang, Honglin

2012

Huang, H. (2012). Improvements to sparse signal processing in compressive sensing and other methods. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/50358>

<https://doi.org/10.32657/10356/50358>

Improvements to Sparse Signal Processing in Compressive Sensing and Other Methods

HUANG HONGLIN

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University

in partial fulfillment of the requirement for the degree of

Doctor of Philosophy

2012

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research done by me and has not been submitted for a higher degree to any other University or Institute.

.....

Date

.....

HUANG HONGLIN

Acknowledgements

First of all, I would like to express my deep and sincere gratitude and thanks to my supervisor, Dr. Anamitra Makur, for his supervision, advice, and guidance on my research, and his encouragement and support through the time of my study towards the Doctor of Philosophy degree. I learned a lot from the inspiring discussions and meetings with him and our group. His knowledge, wisdom, and kindness inspire and enrich my growth as a student and a researcher I want to be.

Secondly, I am very grateful to all my colleagues and friends in Media Technology Lab who have helped me a lot during these years of my study and research at Nanyang Technological University. My special thanks go to Xu Zhiming, Zhang Lei, Achanna Anil Kumar, Divya Venkatraman, Jayachandra Dakala, Asha Vijayakumar, Sathiya Narayanan Sekar, Sujit Kumar Sahoo for their great help in my research. I very appreciate their help over these years.

Thirdly, I want to thank all the faculty members in Division of Information Engineering for providing an excellent education and research environment. And I also would like to thank the technicians of Media Technology Lab.

Finally, I am so grateful to my family for their support and understanding. I would like to give my heartiest appreciation to my parents and Qiu Ya and other family members for their constant support and unconditional love. They always stand by my side to help me in all possible ways. Without their support and love, I could not have completed this thesis.

Abstract

Compressive Sensing (CS), as a newly developed branch of sparse signal processing and representation approaches, has quickly found various applications in a large number of research topics in modern digital signal processing area. This work is devoted to investigating some effective CS and sparse signal processing schemes and approaches, and also to adapting the sparse signal processing idea and CS framework to some specific applications such as object-based surveillance video compression.

In compressive sensing, the sampling strategy and reconstruction algorithms are two major components. In this thesis, we first investigate the optimization of the sampling matrix for effective CS performance. An optimizing method, by using a simple polynomial shrink function and a detector to estimate the convergence to stop the iterations early, is proposed to provide better CS performance. A number of experimental simulations are presented to demonstrate the optimized measurement matrix's effectiveness.

Then, a novel Backtracking-based Adaptive Orthogonal Matching Pursuit (BAOMP) method is proposed to effectively reconstruct or approximate the sparse solutions for CS and other sparse representation problems. Different from other Orthogonal Matching Pursuit (OMP)-type algorithms, the proposed method incorporates a backtracking step to more carefully choose the reliable support set, and at the same time, it does not require the signal's sparsity level to be known before reconstruction. Various experiments on exact sparse signal reconstruction case, noisy signal or noisy measurement approximation case, and two-dimensional (2-D)

compressible signal approximation case are illustrated to show the better performance than that of other known OMP-type methods.

Furthermore, as an application of CS approach and sparse signal processing idea, an object-based surveillance video compression system based on CS is proposed, in which the sparse object error after motion compensation is coded by using CS scheme. In the proposed system, first the front-moving objects are segmented from background, then these are object-based compensated from previous reconstructed frames, and finally the object error blocks are encoded by CS approach and quantized to transmit or store. Extensive experiments show the proposed system's efficiency.

Finally, a simple iterative reconstruction method based on Projection Onto Convex Sets (POCS) is designed to effectively encode the object error. Firstly, we consider the reconstruction performance of one-dimensional (1-D) Autoregressive (AR) source signal, and then use a natural image as the input signal to do some experiments to illustrate the proposed method's performance on image reconstruction. Finally, as an application of the iterative reconstruction method, a novel sparse signal compression scheme based on this iterative method is presented.

Contents

Acknowledgements	i
Abstract	ii
List of Figures	vii
List of Tables	xiii
List of Abbreviations	xiv
1 Introduction.....	1
1.1 Motivation.....	2
1.2 Objective.....	5
1.3 Major Contributions of the Thesis	8
1.4 Organization of the Thesis	10
1.5 Notation	12
2 Review of Sparse Signal Processing.....	13
2.1 Fundamentals of Sparse Signal Representation	13
2.2 Compressive Sensing.....	19
2.3 CS and Sparse Representation Reconstruction Algorithms	22
2.3.1 Convex Relaxation Optimization Algorithms	24
2.3.2 Greedy Reconstruction Algorithms.....	27
2.3.3 Iterative Thresholding Reconstruction Methods	29
2.3.4 Performance Comparison of CS Reconstruction Methods.....	31
2.4 Applications of Compressive Sensing.....	36
3 Optimized Measurement Matrix for Compressive Sensing	38
3.1 Introduction.....	38
3.2 Measurement Matrix and its Properties	40

3.3	Optimization of CS Measurement Matrix	42
3.3.1	Existing Optimization Method	42
3.3.2	The Proposed Optimization Method	43
3.4	Experimental Results and Discussions.....	50
3.4.1	Experiments on Exact Sparse Signal Reconstruction	50
3.4.2	Experiments on Distributed CS Case	55
3.4.3	Experiments on 2-D Compressible Signal Reconstruction	58
3.5	Summary	61
4	Backtracking-based Adaptive OMP Algorithm for CS and Sparse Signal Reconstruction.....	63
4.1	Introduction.....	63
4.2	Backtracking-based Adaptive Orthogonal Matching Pursuit.....	66
4.2.1	Existing Greedy Algorithms.....	66
4.2.2	The Proposed BAOMP Algorithm	70
4.3	Experimental Results and Discussions.....	78
4.3.1	Experiments on 1-D Exact Sparse Signal Reconstruction.....	79
4.3.2	Effect of Parameter Choices on Performance.....	84
4.3.3	Experiments on 1-D Noisy Perturbation Reconstruction	89
4.3.4	Experiments on 2-D Compressible Signal Reconstruction	96
4.4	Summary	99
5	Object-based Surveillance Video Compression using CS	103
5.1	Introduction.....	103
5.2	Object-based Video Compression Scheme	105
5.2.1	Object Segmentation	107
5.2.2	Motion Estimation and Compensation	110
5.2.3	CS-based Object Error Coding.....	111
5.2.4	Quantization	112

5.3	Experimental Results and Discussions.....	113
5.3.1	Experiments on Reconstruction of Single Object Error Block.....	113
5.3.2	Experiments and Comparisons on Whole Video Sequence	118
5.4	Summary	131
6	Iterative Reconstruction of Sparse Signal Representation.....	134
6.1	Introduction.....	134
6.2	Projection Onto Convex Sets Method.....	136
6.3	The Proposed Iterative Reconstruction Scheme.....	137
6.4	Experimental Results and Discussions.....	140
6.4.1	Autoregressive Source Reconstruction Results	140
6.4.2	Image Reconstruction Results	143
6.4.3	Results for Sparse Signal Reconstruction.....	147
6.5	Summary	153
7	Conclusions and Recommendations	155
7.1	Conclusions.....	155
7.1.1	Optimized Matrix for Compressive Sensing	155
7.1.2	Backtracking-based Adaptive OMP Algorithm for CS	156
7.1.3	Video Object Error Coding Method based on CS	157
7.1.4	Sparse Signal Reconstruction based on Iterative Method	158
7.2	Future Works	159
	Author's Publications.....	162
	Bibliography	165

List of Figures

Figure 2.1: The intersection between the ℓ_p -norm ball and the constraint set $\Phi\mathbf{x} = \mathbf{y}$ defines the solution of (P_p) . The intersection is demonstrated here for $p = 2$ (top left figure), $p = 1.5$ (top right figure), $p = 1$ (bottom left figure), $p = 0.7$ (bottom right figure). When $p \leq 1$, the intersection takes place at a corner of the ball, leading to a sparse solution ([81]). **17**

Figure 2.2: A simple MRI image reconstruction example: (a) the original Logan-Shepp phantom test image; (b) sampling domain Ω (Fourier coefficients are sampled along 22 approximately radial lines) in the frequency plane; (c) minimum energy reconstruction obtained by setting unobserved Fourier coefficients to zeros; (d) reconstruction obtained by minimizing the total variation (the reconstruction is exact) ([54]). **26**

Figure 2.3: Schematic diagram of StOMP algorithm ([89]). **28**

Figure 2.4: Mean Square Error (MSE) between the reconstructed signal and the original one (binary sparse signal) as a function of measurement numbers by using different reconstruction algorithms (IHT method is initialized by MP result). **33**

Figure 2.5: The needed measurements to recover signals correctly ($\text{MSE} < 10^{-8}$) as a function of the sparsity level for different methods (IHT method is initialized by MP result). **33**

Figure 2.6: 128×128 image reconstruction results from 8756 measurements: (a) the original image: a part of ‘Lena’, (b) recovered from 1256-term best wavelet approximation PSNR=40.386dB, (c) recovered by BP method, PSNR=40.250dB, (d) recovered by MP method, PSNR=38.661dB, (e) recovered by MP +IHT method, PSNR=38.048dB, (f) recovered by OMP method, PSNR=37.557dB, (g) recovered by StOMP method, PSNR=38.785dB. **35**

Figure 2.7: A Compressive Imaging (CI) camera block diagram. Incident light field (corresponding to the underlying image $\mathbf{x}$) is reflected off a Digital Micromirror Device (DMD) array whose mirror

orientations are modulated in the pseudo-random pattern supplied by Random Number Generators (RNG), each different mirror pattern φ_i produces a voltage at the single photodiode that corresponds to one measurement y_i ([65]). **37**

Figure 3.1: The shrink operation used in the proposed method..... **46**

Figure 3.2: The histograms of the absolute off-diagonal entries of $\mathbf{G}$ before the optimization and afterwards, (a) the initial random matrix (b) matrix optimized by Algorithm 1 with 1000 iterations and $t=0.2$ (fixed threshold mode), $\gamma = 0.5$ (c) matrix optimized by Algorithm 1 with 1000 iterations and t =average of the top 20% of the off-diagonal entries in $\mathbf{G}^n$ (relative threshold mode), $\gamma = 0.5$ (d) matrix optimized by the proposed Algorithm 2 with $\varepsilon = 10^{-8}$ **49**

Figure 3.3: (a) the average reconstruction error as a function of the number of measurements M , by using the initial random matrix and the optimized matrices by Algorithm 1($t=0.2$ (fixed threshold mode) and $t=20\%$ (relative threshold mode)) and Algorithm 2 (b) the corresponding computation times for optimization by the different methods..... **52**

Figure 3.4: (a) the probability of successful reconstruction as a function of the number of measurements M , with the initial random matrix and the optimized matrices by Algorithm 1($t=0.2$ (fixed threshold mode) and $t=20\%$ (relative threshold mode)) and Algorithm 2 (b) the corresponding computation times for optimization by the different methods..... **53**

Figure 3.5: (a) the average reconstruction error as a function of the number of measurements M in DCS case (b) the probability of successful reconstruction as a function of the number of measurements M in DCS case. **57**

Figure 3.6: Reconstructed image from $M = \min(3K, N)$ measurements: (a) the original image, (b) reconstructed image by DCT with the best K -term coefficients, PSNR=37.1907 dB, (c) reconstructed image by using the initial random matrix, PSNR= 32.3009dB, (d) reconstruction error per pixel for (c), (e) reconstructed image by using Elad's algorithm [80], PSNR = 34.6508dB, (f) reconstruction error

per pixel for (e), (g) reconstructed image by using the proposed method, PSNR = 36.2204 dB, (h) Reconstruction error per pixel for (g).....	60
Figure 4.1: Schematic diagrams of the iterations in some OMP-type methods: (a) OMP method (b) SP method (c) the proposed BAOMP method.....	69
Figure 4.2: A sparse signal reconstruction example by using OMP and BAOMP methods.	76
Figure 4.3: Simulations of the exact reconstruction rate vs. the signal sparsity level K by using different methods: (a) Gaussian sparse signal, (b) binary zero-one sparse signal.....	81
Figure 4.4: Simulations of NMSE vs. the signal sparsity level K by using different methods: (a) Gaussian sparse signal, (b) binary zero-one sparse signal.....	82
Figure 4.5: Simulations of the exact reconstruction rate vs. the number of measurements M by using different methods: (a) Gaussian sparse signal, (b) binary zero-one sparse signal.	83
Figure 4.6: Convergence of the proposed method for Gaussian and binary sparse signal when the number of measurements M is fixed.	85
Figure 4.7: Convergence of the proposed method when sparsity level K is fixed.....	87
Figure 4.8: The effect of different parameter choices on the exact reconstruction rate vs. number of measurements M by using the proposed method: (a) Gaussian sparse signal (b) binary sparse signal.	89
Figure 4.9: A reconstruction example for signal perturbation case: (a) original sparse signal (b) original signal perturbed by i.i.d. Gaussian noise $N(0, 10^{-4})$ (c) OMP result, MSE=0.86035 (d) ROMP result, MSE=0.56895 (e) StOMP result, MSE=0.38354 (f) SP result, MSE=0.26936 (g) BP result, MSE=0.20166 (h) BAOMP result, MSE=0.0022514.....	93
Figure 4.10: Simulations of the average reconstruction error for signal perturbation case vs. number of measurements M by using different methods: (a) Gaussian sparse signal (b) binary sparse signal.....	94

Figure 4.11: Simulations of the average reconstruction error for measurement perturbation case vs. number of measurements M by using different methods: (a) Gaussian sparse signal (b) binary zero-one sparse signal. **96**

Figure 4.12: Reconstructed image from $M = \min(\text{round}(3.5K), N)$ measurements: (a) the original image (b) reconstructed image by DCT with the best K -sparse coefficients, PSNR = 32.5091 dB (c) reconstructed image by StOMP method, PSNR = 29.4375dB (d) reconstruction error per pixel for StOMP method (e) reconstructed image by SP method, PSNR = 29.2213 dB (f) reconstruction error per pixel for SP method (g) reconstructed image by GPSR ℓ_1 -norm minimization method, PSNR = 29.8603 dB (h) reconstruction error per pixel for GPSR method (i) reconstructed image by the proposed BAOMP method, PSNR = 31.4372 dB (j) reconstruction error per pixel for BAOMP method. **98**

Figure 5.1: The block diagram of the proposed compression scheme: (a) the proposed encoder (b) the proposed decoder.....**106**

Figure 5.2: The segmented results for one frame of the ‘walk’ video sequence: (a) aggregation of object pixels after step 1 (b) mask after morphological operations after step 2 (c) the final mask after step 3 (d) the segmented objects.....**109**

Figure 5.3: The original object error block segmented from the whole video frame: (a) the whole video frame (b) the resulting object error block.....**115**

Figure 5.4: Reconstruction by BP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16. ...**115**

Figure 5.5: Reconstruction by OMP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16.**115**

Figure 5.6: Reconstruction by StOMP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16.

.....115

Figure 5.7: The reconstruction performance by the proposed method when quantization values Q_t of 16 and 32 are used for ‘walk’ sequence with 30 frames to be coded (a) the Y-plane PSNR values of each reconstructed frame, (b) the bits required for encoding each frame.120

Figure 5.8: The reconstructed frame by using the proposed method and H.264 jm10.2 reference software: (a) the original video frame (b) the reconstructed 5th frame by using H.264 software jm10.2, PSNR=34.438dB, bits=10080bits (c) the reconstructed 5th frame by using the proposed method, PSNR=36.2908dB, bits=5943bits.121

Figure 5.9: The compression performance’s comparison by the proposed method ($Q_t=16$) and h.264 jm10.2 reference software: (a) PSNR values of the whole frame (b) PSNR values of the foreground object (c) PSNR values of the background object (d) the required bits for encoding each frame.124

Figure 5.10: The compression performance’s comparison by the proposed method ($Q_t=4$) and H.264 jm10.2 reference software: (a) PSNR values of the whole frame (b) PSNR values of the foreground object (c) PSNR values of the background object (d) the required bits for encoding each frame.126

Figure 5.11: The compression performance’s comparison for ‘walk’ sequence by the proposed method and run-length coding ($Q_t=8$): (a) PSNR values of the whole frame (b) PSNR values of the foreground object (no transmission lost) (c) PSNR values of the foreground object (20% transmission lost) (d) the required bits for encoding each frame.129

Figure 5.12: The compression performance’s comparison for ‘hall monitor’ sequence by the proposed method and run-length coding ($Q_t=8$): (a) PSNR values of the whole frame (b) PSNR values of the foreground object (no transmission lost) (c) PSNR values of the foreground object (20% transmission lost) (d) the required bits for encoding each frame.....131

Figure 6.1: The iterative reconstruction processing scheme.....	138
Figure 6.2: The convergence rate (%) for three decision rules when $N=8$	142
Figure 6.3: The average iteration number needed for convergence by the three decision rules when $N=8$	142
Figure 6.4: The average reconstruction error of convergence cases for three decision rules when $N=8$	143
Figure 6.5: The reconstructed images by different methods when $P = 10$: (a) the original image, (b) reconstruction by padding the lost coefficients with zeros, PSNR=31.002dB, (c) reconstruction by padding the lost coefficients with zeros and then replacing the corresponding samples with known samples, PSNR=31.741dB, (d) reconstruction by USE rule, PSNR=38.672dB, (e) reconstruction by IE rule, PSNR=41.396dB, (f) reconstruction by KSE rule, PSNR=41.385dB.....	145
Figure 6.6: The PSNR values of reconstructed images for different P	146
Figure 6.7: Convergence plots of the iteration method for image reconstruction.	147
Figure 6.8: Reconstruction results: (a) the original object error block (b) reconstruction by 10 iterations, PSNR=34.958dB (c) reconstruction by 100 iterations, PSNR=35.427dB (d) reconstruction by 1000 iterations, PSNR=36.040dB.	149
Figure 6.9: The modified iterative reconstruction processing scheme.	150
Figure 6.10: Reconstruction results with quantization: (a) reconstruction with quantization value 8 by 10 iterations, PSNR=34.189dB (b) reconstruction with quantization value 16 by 10 iterations, PSNR=33.505dB.	150
Figure 6.11: The PSNR values changing with iterations: (a) no quantization is used (b) quantization value 8 is used (c) quantization value 16 is used.....	151

List of Tables

Table 3.1: The optimization method proposed by Elad [80]	45
Table 3.2: The proposed method for optimizing CS measurement matrix.....	47
Table 4.1: The proposed Backtracking-based Adaptive OMP method	72
Table 5.1: The Performance Comparison of BP, OMP and StOMP Methods.....	116

List of Abbreviations

1-D	one-dimensional
2-D	two-dimensional
ADC	Analog-to-Digital Conversion
AR	Autoregressive
BAOMP	Backtracking-based Adaptive OMP
BP	Basis Pursuit
BPDN	Basis Pursuit De-Noising
CoSaMP	Compressive Sampling Matching Pursuit
CS	Compressive Sensing
DCS	Distributed Compressive Sensing
DCT	Discrete Cosine Transform
DS	Dantzig Selector
FOCUSS	FOCal Underdetermined System Solver
IDCT	Inverse-DCT
IE	Iteration Error
IHT	Iterative Hard Thresholding
IRLS	Iteratively-Reweighed-Least-Squares
IST	Iterative Soft Thresholding
KSE	Known Sample Error
LP	Linear-Programming
MC	Mutual Coherence
MDC	Multiple Description Coding
MP	Matching Pursuit
MPEG	Moving Picture Experts Group
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
NMSE	Normalized Mean Square Error
NP	Non-deterministic Polynomial-time

OMP	Orthogonal Matching Pursuit
PCG	Preconditioned Conjugate Gradients
POCS	Projection Onto Convex Sets
PSNR	Peak Signal-to-Noise Ratio
RIP	Restricted Isometry Property
ROMP	Regularized OMP
SP	Subspace Pursuit
StOMP	Stagewise OMP
SVD	Singular Value Decomposition
TOMP	Tree-based OMP
TV	Total Variation
USE	Unknown Sample Error
WT	Wavelet Transform

Chapter 1

Introduction

The field of sparse signal processing deals with a signal that has a sparse representation on some bases or over-complete dictionaries, with a linear combination of a small number of elementary functions called atoms [1-8]. Sparseness has a wide application in signal processing field, and most techniques exploit the sparseness property to more effectively process the underlying signal [9]. For example, sparseness is one reason for the extensive use of some popular transforms and frames such as Discrete Cosine Transform (DCT), Wavelet Transform (WT) [10-12], Ridgelet Transform [13], Contourlet Transform [14-15], Bandelet Transform [16-17], and Curvelet Frames [18-19] in practical signal processing schemes. These transforms and frames are often used to reveal certain structures of the given signals and to represent them in sparse and compact representations. Hence, as a research focus in signal processing domain, sparse representation has increasingly been applied in many application areas, such as signal and image denoising [20-28], image restoration [29-33], image deblurring [34], image inpainting [35-37], image deconvolution [38-40], image super-resolution [41-42], image compression [43-46], blind source separation [47-50], image decomposition and analysis [51-53], and more. For example, the sparseness property is the foundation of wavelet soft-thresholding denoising approach [20]. Another more useful example is signal compression [43-46].

Firstly the signal is transformed into a sparse representation, and then some compression strategy is used to encode the few significant coefficients and discard other smaller ones, hence the transmitted or stored bitrates can be effectively reduced.

This thesis mainly aims to investigate some effective sparse signal processing approaches to deal with a sparse signal or a compressible signal that is sparse on some basis or dictionary. Particularly, the branch of sparse signal processing considered in this work is CS approach, which is a newly emerged technique that aims to acquire sparse and compressible signals at close to their intrinsic information rate rather than their Nyquist rate [54-57]. In the next subsections, firstly the motivation and objective are explained, then the major contributions of the thesis are outlined, and finally the organization and used notations are established.

1.1 Motivation

The technique of finding a sparse representation with few significant coefficients is often referred to as sparse coding. If the locations of significant coefficients are known in advance, the decoding procedure is simple, which merely requires the summation of all the relevant atoms, with appropriate weights. However, unlike the transform coder which has a direct linear invertible transform such as wavelet transform, the reconstruction of the sparse representation with an over-complete dictionary is not trivial. It extends the linear representation idea to consider more flexible redundant representation (called dictionaries) where the linear analysis transform is replaced by a nonlinear sparse representation operator. The general problem of finding a representation with the smallest number of atoms from an

arbitrary dictionary is Non-deterministic Polynomial-time (NP)-hard [1, 2, 54-56]. In the existing literature there are considerable popular methods which attempt to solve this problem. These algorithms include the greedy algorithms [1, 58-60], convex relaxation optimization methods [2, 3, 61], and some iterative thresholding algorithms [62-63]. The greedy algorithms iteratively build up the sparse signal approximation one coefficient at a time, e.g. Matching Pursuit (MP) method [1]. The convex relaxation optimization method relaxes the non-convex ℓ_0 -norm problem by some convex functions, and processes all the coefficients simultaneously to give a globally optimized solution, like Basis Pursuit (BP) method [2-3]. Besides these two efficient sparse reconstruction algorithms, iterative thresholding method provides an alternative approach to regularize inverse problems by sparsity constraints [62-63].

Compressive sensing is a recently developed branch that is originated from sparse and redundant representations, and fast became a center of interest of its own [54-57]. It provides a new way of thinking about how to acquire and code signals in the most efficient manner. Exploiting the sparseness of the underlying signal, their sampling can be made far more effective compared to the classical Nyquist-Shannon sampling theorem. The impact of this technique is immense, and the growing number of applications is being explored across a range in various disciplines and research fields, including compressive imaging [64-65], medical imaging [66-68], Analog-to-Digital Conversion (ADC) [69-71], geophysical data analysis [72], compressive radar [73-75], communications [76-78], and so on.

The compressive sensing idea asserts that if one simply projects a sparse or compressible signal down to a much smaller observation space, by using an

appropriate observation matrix, then the nonlinear reconstruction techniques developed for building sparse representations can be used to reconstruct the signal. Furthermore, the smaller space has been shown both in terms of the number of samples [54, 56] and the number of bits required in encoding the samples [79].

There are two major components to CS: the sampling strategy and the reconstruction algorithms. While conventional Nyquist sampling involves measuring a signal at regular interval to satisfy the Nyquist rate, the concept of sampling in CS is much more general. Sampling in CS consists of making a random linear projection of the signal into a low dimensional space, and researchers have found that the CS idea can often be used in most conventional sampling scenarios. For example, in Magnetic Resonance Imaging (MRI) system, the scanners sample lines within the spatial Fourier domain of the image, and CS technique can be used for MRI with randomized trajectories or even deterministic trajectories (sampling a small number of radial lines) in the Fourier space to reduce the sampling measurements and shorten the scan time.

One key difference between conventional sampling scheme and compressive sensing approach is that the reconstruction operator for CS is nonlinear. Essentially, CS selects the significant coefficients for the underlying signal in some sparse representation and then computes a least squares approximation using the associated sampling basis function. However, finding the significant coefficients is a combinational search problem and practically cannot be solved directly. Instead of the combinational problem, much of CS theory has concentrated on proving that near optimal performance is possible by using either a convex relaxation algorithm that

solves a linear or quadratic program, or a greedy algorithm that iteratively selects the coefficients in a greedy way one at a time or in groups.

Even though there are many theoretical foundations to the CS framework, and there also exist considerable tools and algorithms for actual applications in signal processing that benefit from it, this research area is still quite young, which will need more effort to address many fundamental questions. The main questions include: 1) Investigate the design and learning of good sampling matrix and their characteristics; 2) Develop and implement novel fast sparse reconstruction and approximation algorithms for CS problems and other sparse processing issues; 3) Analyze the theoretical and practical performance of the CS approach when it is applied to practical signal processing applications; 4) Design signal processing applications based on the CS idea, and so on.

This thesis is an effort towards designing more effective CS and other sparse reconstruction algorithms and approaches, and also considering the practical CS performance when it is used in object-based video compression, which are basically the motivation of writing this thesis.

1.2 Objective

Compressive sensing, as a newly emerged scheme over the last five years from theoretical work developed within the field of sparse signal representation, quickly became a research focus in various research areas. Sparse signal representation extends the transform-based representation idea considering more flexible redundant

representations where the linear analysis transform is replaced by a nonlinear sparse representation operator.

Motivated by the rapid development of CS framework in theoretical foundations and applications, the works dedicated in this thesis mainly focus on the following objectives for sparse signal processing and CS applications:

- **Improve CS and sparse signal reconstruction performance by**

optimizing the measurement matrix: one fundamental problem in CS

is how to construct a measurement matrix Φ so that the number of measurements is near minimal. Currently, most work on CS assumes that Φ is drawn at random from some probability distribution, which simplifies the theoretical analysis, and also facilitates a simple implementation [54-57, 79].

To improve the CS reconstruction performance (e.g. increasing the probability of successful reconstruction, reducing the average reconstruction error), a simple polynomial function is used as the shrink operation to down scale the inner products between columns of the given sampling measurement matrix.

Sampling by this optimized matrix, CS performance is improved greatly.

Moreover, the optimized matrix may also be used in Distributed Compressive Sensing (DCS) case to further improve the reconstruction performance.

- **Design and implement a novel backtracking-based adaptive**

orthogonal matching pursuit algorithm for CS and other

sparse reconstruction problems: to deal with CS and other sparse

signal reconstruction and approximation problems, a BAOMP method is proposed. As an extension of OMP method, the proposed BAOMP method

incorporates a backtracking technique to check the previous chosen atoms' reliability and then delete the unreliable atoms at each iteration. Furthermore, by using two preset thresholds to control the numbers of added and deleted atoms at each time, the proposed method does not require the sparsity level to be known a priori, but still can give better performance than other OMP-type methods, whether for exact sparse signal reconstruction case, or for noisy perturbation case, or for 2-D compressible signal case.

- **Implement an efficient video compression scheme based on coding the video object error by CS framework:** with the theoretic development of sparse representation and CS framework, the research trend has also rapidly explored the concept of CS to be used in practical signal applications. Hence, design and implementation of the applications exploiting the ideas of sparse signal representation and CS becomes another major objective in these areas. For example, for signal denoising, the sparse representation model can be adapted in real processing settings to more effectively solve the underlying problems [20-28]. These successful applications motivate us to investigate the use of the sparse property and CS idea in the object-based video compression to encode the sparse object error blocks. In order to effectively compress the object error blocks, an object error coding method based on CS is proposed in this thesis.
- **Propose a POCS iterative reconstruction method to effectively code the sparse object error block:** to effectively code the object error blocks generated in segmentation-based video compression scheme, a simple

iterative method based on POCS is designed. Some experiments show it could correctly and effectively reconstruct the underlying sparse signals.

In this work, the above objectives of optimizing sampling matrix and designing useful reconstruction algorithm have been briefly explained. Moreover, we also manage to adapt the sparse signal representation idea and CS framework to the specific surveillance video compression application, which provides other objective of designing applications based on CS approach.

1.3 Major Contributions of the Thesis

In this thesis, we study some sparse signal processing problems, especially the CS problems, in terms of the optimized sampling matrix, the efficient reconstruction algorithm and the applications in object-based video compression. The main contributions are presented in Chapters 3, 4, 5 and 6. In Chapter 2, to make the thesis self-contained, some fundamentals and the current state-of-art reconstruction algorithms for sparse representation and CS are reviewed from existing literature. Chapter 7 concludes the thesis. The most important individual contributions are:

- In Chapter 3, we study the optimized measurement matrix for better CS reconstruction performance. Based on the method proposed by Elad [80], we modified this scheme by using a simple polynomial function to down scale the inner products between columns of sampling matrix, and also using a detector to estimate the convergence to early stop the iterations when necessary. Starting with an initial random measurement matrix, the proposed method could simply reduce the average mutual coherence of the random sampling

matrix to obtain a better-performance sampling matrix. The experiments reveal that this optimized matrix leads to better CS reconstruction performance with low extra computational cost. We also explore the performance of the optimized matrix when it is used in a simple Distributed CS situation.

- In Chapter 4, we propose a BAOMP algorithm to reconstruct or approximate the sparse solution for CS problem. We take into account different application cases, such as exact sparse signal reconstruction, noisy signal or noisy measurement approximation, and 2-D compressible signal approximation (like natural images). The proposed method incorporates a backtracking step to more accurately choose the reliable atoms, and moreover, it does not require the sparsity level K a priori to assist the reconstruction. This property may be helpful since in some applications (such as blind sparse signal processing) this information may be unknown before reconstruction.
- In Chapter 5, we present a new object-based surveillance video compression system, in which the segmented object error blocks after motion compensation are coded by using CS scheme. The proposed system first assumes the moving objects have been segmented from background and object-based motion compensated from the previous reconstructed frames, and then the resulting object error blocks are encoded by using some CS random matrix. Finally the collected CS measurements can be quantized to store or transmit. At the decoder side, the system does an inverse processing.

-
- In Chapter 6, a simple iterative method based on POCS scheme is proposed to encode the sparse object error block. After the same processing as that in above video compression scheme based on CS idea, the resulting object error blocks can be effectively encoded by the proposed iterative reconstruction method. From partial time-domain samples and partial frequency-domain coefficients, the proposed method iteratively projects the signal onto the known samples' subspaces and the known coefficients' subspaces. Finally, experiments show the underlying signal could be recovered closely.

1.4 Organization of the Thesis

As explained above, the objective of this thesis is to provide effective sparse signal processing methods and their applications in practical signal processing areas. To achieve this objective, the remainder of the thesis is organized as follows:

In Chapter 2, some fundamentals of sparse signal processing and CS techniques are reviewed from existing literature to make the thesis self-contained. After that, three kinds of popular algorithms for sparse signal representation and CS reconstruction are also briefly reviewed. Finally, a performance comparison of these reconstruction algorithms is provided.

Chapter 3 presents the optimized construction of the measurement sampling matrix for CS. Firstly, the concepts of Restricted Isometry Property (RIP) and Mutual Coherence (MC) are briefly reviewed, which provide sufficient constraints on the measurement matrix to make it work well for CS problems. Then the proposed optimization method is described in detail. Finally, experiments on exact sparse signal

reconstruction are simulated and compared with other methods. Moreover, we also do some simulations on DCS case to further investigate the performance.

Chapter 4 proposes a Backtracking-based Adaptive OMP algorithm to improve the performance of sparse signal approximation and CS reconstruction. Firstly, the property and detailed description of the proposed BAOMP method is presented. Then we use the proposed method to do some experiments on exact sparse signal reconstruction, noisy signal or noisy measurement approximation, and 2-D compressible signal approximation cases. Furthermore, we also analyze the effect of parameter choices on computational complexity and performance. We also compare the proposed BAOMP method with other state-of-art CS reconstruction algorithms and show the performance improvement.

In Chapter 5, CS approach is adapted to an object-based surveillance video compression system to encode the object error after segmentation and motion compensation, which is usually very sparse in the segmented block and makes CS technique appropriate to be used. Firstly, the whole surveillance video compression scheme based on CS is presented in detail. Then some simulations on reconstruction of a single object error block are provided. Finally, we also do some experiments on a surveillance video sequence and then compare the performance with H.264 and run-length coding to show its efficiency.

Chapter 6 presents a POCS-based iterative reconstruction method for the sparse object error block compression. The object error blocks are segmented as the above one described in Chapter 5. In this chapter, firstly the proposed iterative reconstruction method is presented. Then the simulations on 1-D AR source signal,

natural image and object error blocks are provided and discussed. This method provides an alternative to consider how to use the sparse representation property to effectively compress the video signal.

Finally, in Chapter 7, the results obtained and presented in this thesis are concluded. Some possible extensions and challenges for future research in sparse signal processing and CS areas, and some other open problems are also briefly outlined and discussed.

1.5 Notation

This section introduces some notations that are widely used in this thesis. Bold-faced quantities denote matrices and vectors, such as in Φ , $\mathbf{x}$, $\mathbf{y}$, etc. They may be followed by some subscripts to denote their dimensions. For example, $\Phi_{M \times N}$ denotes a matrix with M rows and N columns. The subscripts will generally be omitted if they are clear from the context. Φ^T , Φ^+ and Φ^{-1} denote the transpose, the pseudo-inverse, and the inverse of matrix Φ , respectively.

The symbol $\mathbb{R}$ denotes the set of real numbers. Sometimes the superscript is used to denote the dimension of numbers, such as $\mathbb{R}^N$ denotes a real-value signal space with dimension- N . The Greek symbols Λ and Γ denote the index sets of atoms chosen from some matrix with each column as an atom.

Chapter 2

Review of Sparse Signal Processing

This chapter reviews some basic definitions and important concepts about sparse signal processing, especially the branch of CS approach, which are used extensively throughout the thesis. In addition, we also introduce some state-of-art reconstruction algorithms for solving the sparse signal representation and CS problems.

2.1 Fundamentals of Sparse Signal Representation

The technique of finding a sparse representation with few significant coefficients is often referred to as Sparse Coding. The heart of a sparse signal processing model is a simple linear system of equations. We consider a matrix $\Phi \in \mathbb{R}^{M \times N}$ with $M < N$, and define the underdetermined linear system of equations $\Phi \mathbf{x} = \mathbf{y}$. This system has more unknowns than equations, and it has either no solution or infinitely many solutions. Avoiding the case of having no solution, we shall hereafter assume that Φ is full-rank, implying that its columns span the entire space $\mathbb{R}^M$ [81]. Clearly, since there are infinite possible solutions to explain the measurement vector $\mathbf{y}$, some questions will be asked: which one is the best solution? And how can we find the proper solution $\mathbf{x}$?

A familiar way to do this is regularization, where a function $J(\mathbf{x})$ that evaluates the desirability of a would-be solution $\mathbf{x}$ is introduced, with smaller values being preferred. Hence, we get the general optimization problem (P_J)

$$(P_J): \quad \min_{\mathbf{x}} J(\mathbf{x}) \quad \text{subject to} \quad \mathbf{y} = \Phi \mathbf{x} \quad (2.1)$$

In this optimization problem (P_J), $J(\mathbf{x})$ will govern the solution that we may obtain.

The most known choice of $J(\mathbf{x})$ is the squared Euclidean norm $\ell_2 = \|\mathbf{x}\|_2^2$. This problem (P_2) results in a unique solution $\hat{\mathbf{x}} = \Phi^T(\Phi\Phi^T)^{-1}\mathbf{y} = \Phi^+\mathbf{y}$ - the so-called minimum-norm solution. Due to its simplicity as manifested by the above closed-form unique solution, the use of ℓ_2 -norm is widespread in various engineering fields to solve most inverse problems. However, ℓ_2 -norm is not always the best choice in some cases, such as the sparse representation problems. Hence there are many other better choices for $J(\mathbf{x})$. Actually, any strictly convex function $J(\mathbf{x})$ guarantees the above (P_J) problem has a unique solution [81].

The sparse signal representation considers the problem (P_0) obtained from the general problem (P_J) with choosing $J(\mathbf{x}) = \|\mathbf{x}\|_0 \equiv \lim_{p \rightarrow 0} \|\mathbf{x}\|_p^p = \#\{i: x_i \neq 0\}$, that is

$$(P_0): \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{y} = \Phi \mathbf{x} \quad (2.2)$$

However, due to the discrete and discontinuous nature of ℓ_0 -norm, the standard convex analysis ideas which underpin the analysis of (P_2) do not apply. Solving (P_0) problem is a classical combinatorial search problem. That is, one has to exhaustively search through all possible subsets to get the corresponding subsystems $\mathbf{y} = \Phi_{\Lambda} \mathbf{x}_{\Lambda}$ where Φ_{Λ} denotes the sub-matrix having the columns chosen from those columns of

Φ with indices in Λ , and to check if $\mathbf{y} = \Phi_{\Lambda} \mathbf{x}_{\Lambda}$ is satisfied. The complexity of exhaustive search is exponential in N , and therefore (P_0) is in general NP-hard.

Even though the choices of other strictly convex function $J(\mathbf{x})$ may not lead to a closed-form solution, the fact that they are strictly convex implies a unique solution. Special cases of interest for convex functions are all the ℓ_p -norms for $p \geq 1$, which are defined as

$$\|\mathbf{x}\|_p^p = \sum_i |x_i|^p \quad (2.3)$$

In particular, the ℓ_{∞} -norm considers the maximal entry of the vector $\mathbf{x}$, and the ℓ_1 -norm sums all the absolute entries. Our special interest is the ℓ_1 -norm due to its tendency to sparsify the solution.

The choice $J(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum |x_i|$ is convex but not strictly so, thus the problem

$$(P_1): \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{y} = \Phi \mathbf{x} \quad (2.4)$$

may have more than one solution. However, the problem (P_1) can be easily converted to a linear programming structure and the ℓ_1 -norm tends to prefer sparse solution.

Consider the unknown vector $\mathbf{x}$ is replaced by $\mathbf{x} = \mathbf{u} - \mathbf{v}$ in (P_1) , where $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$ are both non-negative vectors such that $\mathbf{u}$ takes all the positive entries in $\mathbf{x}$, other entries are zero, and $\mathbf{v}$ takes all the negative entries in $\mathbf{x}$, other entries are zero. By denoting $\mathbf{z} = [\mathbf{u}^T, \mathbf{v}^T]^T \in \mathbb{R}^{2N}$, it is easy to see that $\|\mathbf{x}\|_1 = \mathbf{1}^T(\mathbf{u} + \mathbf{v}) = \mathbf{1}^T \mathbf{z}$ (here the notation $\mathbf{1}$ stands for a vector of ones with the proper length.) and $\Phi \mathbf{x} = \Phi(\mathbf{u} - \mathbf{v}) = [\Phi, -\Phi] \mathbf{z}$. Hence, the optimization problem (P_1) becomes

$$\min_{\mathbf{z}} \mathbf{1}^T \mathbf{z} \quad \text{subject to} \quad \mathbf{y} = [\Phi, -\Phi] \mathbf{z} \quad \text{and} \quad \mathbf{z} \geq 0 \quad (2.5)$$

In this way, the ℓ_1 -norm minimization problem can be solved as a classical Linear-Programming (LP). The problem (P_1) is equivalent to a LP problem.

Moving from ℓ_2 -norm regularization to ℓ_1 -norm promotes a sparser solution. Actually, ℓ_p -norms with $p < 1$ also produce sparse solutions. ℓ_p -norms are no longer formal norms when $p < 1$, however, we use the term norm for these functions as well for simple description. The tendency of ℓ_p -norm to drive sparse solution can be illustrated in the following way: consider the ℓ_p -norm regularization problem

$$(P_p): \quad \min_{\mathbf{x}} \|\mathbf{x}\|_p^p \quad \text{subject to} \quad \mathbf{y} = \Phi \mathbf{x} \quad (2.6)$$

The constraint set of linear equations defines a feasible set of solutions that appears as a hyperplane of dimension $\mathbb{R}^{N-M}$ embedded in the $\mathbb{R}^N$ space. Geometrically speaking, as shown in Figure 2.1, solving (P_p) is done by blowing a ℓ_p -norm balloon centered around the origin, and stopping its inflation when it first touches the feasible set. Figure 2.1 presents a simple demonstration of the process for a tilted hyperplane (the constraint set) and several p values: 2, 1.5, 1 and 0.7. One can see the ℓ_p -norms with $p \leq 1$ tend to give the intersection points at the ball-corners, which take place on the axes to generate sparse solutions. As opposed to this, ℓ_2 -norm and even $\ell_{1.5}$ -norm give intersection points which are not sparse.

Another similar measure promoting sparsity is the weak ℓ_p -norm. Denoting the number of entries in $\mathbf{x}$ exceeding ε by $\mathbf{N}(\varepsilon, \mathbf{x})$, the weak ℓ_p -norm is defined as

$$\|\mathbf{x}\|_{w\ell_p}^p = \sup_{\varepsilon > 0} \mathbf{N}(\varepsilon, \mathbf{x}) \cdot \varepsilon^p \quad (2.7)$$

Here, $0 < p \leq 1$ is the interesting range. This weak ℓ_p -norm is a popular measure of sparsity used in the mathematical analysis community. Actually, the above two ℓ_p -norms become the same for a vector with uniform energy spread among all the non-zero entries. Although the ℓ_p -norms with $0 < p < 1$ give very powerful sparsity measure, each choice of $0 < p < 1$ leads to a non-convex optimization problem. However, if sparsity is a desired property, ℓ_p -norm with $0 < p \leq 1$ serves it well.

Figure 2.1: The intersection between the ℓ_p -norm ball and the constraint set $\Phi\mathbf{x} = \mathbf{y}$ defines the solution of (P_p) . The intersection is demonstrated here for $p = 2$ (top left figure), $p = 1.5$ (top right figure), $p = 1$ (bottom left figure), $p = 0.7$ (bottom right figure). When $p \leq 1$, the intersection takes place at a corner of the ball, leading to a sparse solution ([81]).

There are other regularization functions of $\mathbf{x}$ that promote sparsity just as well as the ℓ_p -norm. In fact, any function $J(\mathbf{x}) = \sum_i \rho(x_i)$ with $\rho(x)$ being symmetric, monotonically non-decreasing, and with a monotonic non-increasing derivative for $x \geq 0$ will serve the purpose of promoting sparsity [81]. Some classic examples are $\rho(x) = 1 - \exp(-|x|)$, $\rho(x) = \log(1 + |x|)$ and $\rho(x) = |x|/(1 + |x|)$.

At present, finding sparse solutions to underdetermined linear systems has become better behaved and well developed in signal processing. It has been found that many media types such as image and video can be sparsely represented efficiently by using dictionary and frame representations, and in fact many important tasks in these areas can be viewed as finding sparse solutions to underdetermined systems of linear equations. For example, Curvelet frame [18-19] and other non-decimated transforms [13-17] are the representations that lead to be sparse.

The field of sparse and redundant representations is relatively young. The early core ideas appeared in a pioneering work by Stephane Mallat and Zhifeng Zhang in 1993 [1], with the introduction of the idea of dictionaries to replace the traditional critically-sampled wavelet transform. Their work put forward some core ideas such as using a greedy pursuit (matching pursuit) technique to approximate a sparse solution to an underdetermined linear equation system, and characterizing the property of dictionaries by their coherence measure, and more. A second important contribution was by Scott Shaobing Chen, David Donoho, and Michael Saunders [2-3], who introduced another pursuit technique that uses the ℓ_1 -norm for regularizing sparsity. They show that the quest for sparsest solution could be tackled as a convex programming task and a basis pursuit method was used to get the proper solution.

After that, there are many deeper analyses of these algorithms and their applications in various disciplines and research areas, including novel greedy algorithms and their analysis [58-60, 82-93], convex relaxation methods and their analysis [54, 94-105], and applications such as signal and image denoising [20-28], image restoration [29-33], image deblurring [34], image inpainting [35-37], image deconvolution [38-40], image super-resolution [41-42], image compression [43-46], blind source separation [47-50], image decomposition and analysis [51-53], and so on.

2.2 Compressive Sensing

As a newly developed branch of sparse signal processing and representation, compressive sensing received great attention in the last few years. Originated from sparse representation idea, it fast became a center of interest of its own [54-57]. It provides a new approach of thinking about how to acquire and code signals in the most efficient way. Exploiting the sparseness of the underlying signal, their sampling can be made far more effectively compared to the classical Nyquist-Shannon sampling theorem. This idea emerged in 2006 from Emmanuel Candès, Justin Romberg, and Terence Tao [54-55], David Donoho [56] and others followed, and the theory and practice of this framework were beautifully formed and quickly became popular, sweeping many researchers and various research areas.

The well-known Shannon sampling theory states that to avoid losing information when capturing a signal, the sampling must be at least two times the signal bandwidth. In many applications, including digital image and video cameras, the Nyquist rate is so high that too many samples are taken, making compression a

necessity prior to storage or transmission. The traditional mode of data acquisition is to first uniformly sample the signal (at or above its Nyquist rate), and then to use some source coding methods for compression, such as transform-based coding. The fact that a compressible signal can be well approximated by a K -sparse signal representation forms the foundation of transform-based coding [10]. In transform-based compression, the raw data samples are first transformed to a sparse representation in some basis, and then the largest K coefficients are kept to encode while the other smaller coefficients are discarded, thereby reducing the amount of data required to be stored or transmitted.

Compressive sensing is a new data acquisition and compression technique that aims to measure sparse and compressible signals at close to their intrinsic information rate rather than their Nyquist rate [54-57]. It is based on the revelation that a small group of non-adaptive linear projections of a sparse signal contains enough information for reconstruction. Like the traditional transform-based compression scheme exploiting the fact that natural signal may be represented sparsely in an appropriate orthonormal basis (e.g., wavelet) where most of the coefficients are zero or small and only a few are large, CS also exploits a signal's sparse property, and states that a signal with a sparse representation can be recovered from a small number of randomly projected measurements, provided an appropriate nonlinear recovery scheme is used. However, rather than directly measuring the original signal, CS measures a signal that is a linear combination of the components of the original signal. By using the non-adaptive linear projections, the collected CS measurements preserve the structure of the given signal, and then the original signal can be reconstructed from these measurements using some sparse signal reconstruction algorithms.

The reconstruction processing of CS considers the same optimization problem (P_0) (2.2) as the reconstruction of over-complete sparse signal representation. In CS framework, the M linear measurements $\mathbf{y} = \Phi \mathbf{x}$ of the length- N signal $\mathbf{x}$ are collected by an $M \times N$ sampling matrix Φ with $M < N$. That means y_i are computed as the inner products between $\mathbf{x}$ and a collection of vectors $\{\phi_i\}_{i=1}^M$. (For multi-dimensional data, it can be vectored into a long 1-D vector. So in this thesis just the 1-D case is considered.)

Similar to over-complete sparse signal representation, the reconstruction of $\mathbf{x}$ from $\mathbf{y}$ is generally ill-posed. However, Candès [54-55, 106-107] and Donoho's [56] recent work states that if $\mathbf{x}$ is sparse enough, in the sense that there are very limited non-zero components in $\mathbf{x}$, then exact recovery is possible and the 'true' signal $\mathbf{x}$ actually is the solution to a simple convex optimization problem. It was proved in [54] that when $M \geq c \cdot K \cdot \log N$, $\mathbf{x}$ is the unique solution to the above convex optimization (P_1) . Here K is the sparsity level, M is the number of measurements, N is the signal length, and $c > 0$ is some small constant value, which depends on the desired probability of successful reconstruction. The core tenet of CS is that it is unnecessary to measure all the N values of a length- N sparse signal; rather, one can recover the signal using a small number of linear projections onto an incoherent basis [54, 56].

For a compressible signal $\mathbf{x}$ or the case when noise perturbation exists, the exact constraint $\mathbf{y} = \Phi \mathbf{x}$ is often relaxed, which will lead to a error-tolerant version of the optimization problem (P_0) , that is

$$(P_0^\epsilon): \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{y} - \Phi \mathbf{x}\|_2 \leq \epsilon \quad (2.8)$$

with error tolerance $\epsilon > 0$. More particularly, if $\mathbf{x}$ is compressible on a basis Ψ , the above problem (P_0^ϵ) can be adapted to

$$(P_0^\epsilon)_\Psi: \quad \min_{\mathbf{x}} \|\Psi^T \mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{y} - \Phi \mathbf{x}\|_2 \leq \epsilon \quad (2.9)$$

It has been shown that if the expansion coefficients $\Psi^T \mathbf{x}$ are compressible and at least $M = O(K \log(\frac{N}{K}))$ Gaussian random measurements are made, the solution to (2.9) has error of the same order as the approximation of the best K non-zero coefficients [107].

2.3 CS and Sparse Representation Reconstruction Algorithms

As discussed in Chapter 1, CS approach can work based on two important issues: the sampling strategy and the reconstruction algorithms. While conventional Nyquist sampling involves measuring a signal at regular interval to satisfy the Nyquist rate, the concept of sampling in CS is much more general. Sampling in CS consists of making a random linear projection of the signal into a low dimensional space. This sampling scheme is feasible based on an assumption that the sampling matrix Φ must satisfy a sufficient condition called RIP Property. Fortunately, Donoho states that nearly any matrix generated at random from a uniform probability distribution can be a measurement sampling matrix [57]. The key is that with high probability, Φ will be a sampling from a basis that is incoherent with the signal sparse-including basis. For 1-D signals, Φ was usually taken to be a Gaussian i.i.d. matrix [54]. However, for 2-D images, N can be fairly large (of the order of $10^4 - 10^6$), which

makes the storage and computation of a Gaussian ensemble very difficult. Thus for easy storage and computational efficiency, [97] suggested to apply a partial random Fourier matrix, i.e., Φ takes the form of $\Phi = F_\Omega$, where F_Ω represents a $M \times N$ matrix from random selection of M rows out of the $N \times N$ Fourier matrix. In [57], a multi-scale CS approach was proposed, where different scales of wavelet coefficients are segregated and sampled with different matrix ensembles. Other possible choices of Φ can be found in [57, 89].

In all, the common choices for measurement matrix Φ include:

- 1 **Gaussian ensemble**: each entry of Φ_{ij} is randomly generated from i.i.d. $N(0, \frac{1}{M})$ Gaussian distribution;
- 2 **Random signs ensemble**: Each entry of Φ_{ij} has value of $\pm 1/\sqrt{M}$ with signs chosen independently and both signs equally likely;
- 3 **Partial Fourier ensemble**: randomly select M rows out of the $N \times N$ Fourier matrix, getting a $M \times N$ partial Fourier matrix;
- 4 **Partial Hadamard ensemble**: randomly select M rows out of the $N \times N$ Hadamard matrix, getting a $M \times N$ partial Hadamard matrix (only consider $N = 2^k$ case).

Except the sampling matrix should satisfy RIP property or other constraint, a computationally tractable algorithm should also be needed to recover the signal from the under-sampling measurements. CS reconstruction problem can be viewed as a sparse approximation problem, that is, the problem of approximating a signal with the best linear combination of K elements from a redundant dictionary Φ . At present,

there exist three kinds of major sparse reconstruction algorithms that are effective for CS and other sparse signal reconstruction problems. They are convex relaxation optimization algorithms [94-105], greedy algorithms [26, 27, 30-33] and iterative thresholding methods [62-63, 97, 108-110]. In the following subsections, we will give some brief reviews on each of the reconstruction methods.

2.3.1 Convex Relaxation Optimization Algorithms

The convex relaxation optimization method relaxes the non-convex ℓ_0 -norm problem (P_0) by some convex functions, and processes all the coefficients simultaneously to give a global optimization solution, like BP method [2-3]. Besides BP method, there are many other convex relaxation methods, such as Total Variation (TV) minimization method [54], FOCal Underdetermined System Solver (FOCUSS) [99], Dantzig Selector (DS) [100], Lasso method [101-102], Basis Pursuit De-Noising (BPDN) method [2-3], and so on. In the following part, we briefly introduce some convex relaxation methods.

Instead of considering the ℓ_0 -norm problem (P_0) (2.2), BP method solves an almost everywhere differentiable and often convex ℓ_1 -norm optimization problem (P_1) (2.4). As described in [2, 3], the ℓ_1 -norm optimization problem can be converted into a linear programming problem as shown in equation (2.5) which can be implemented easily by using some standard methods such as interior-point methods, named as Basis Pursuit. The equivalence of ℓ_0 -norm and ℓ_1 -norm minimizations could be shown under some conditions, and BP method states that ℓ_1 -norm optimization can exactly recover K -sparse signals and accurately approximate

compressible signals with a probability exceeding $1 - O\left(N^{-\frac{\rho}{\alpha}}\right)$ (here $\rho > 0$ is a universal constant, $\alpha > 0$ is a sufficiently small constant value) using only $M \geq \alpha \cdot K \log N$ Fourier sampling measurements [54, 111-112]. For random sampling Gaussian measurements, $M = O(K \log(N/K))$ measurements are sufficient for accurate reconstruction with high probability [56].

Total variation minimization method [54] solves the following problem to impose a sparse constraint,

$$(P_{TV}): \quad \min_{\mathbf{x}} \|\mathbf{x}\|_{TV} \quad \text{subject to } \mathbf{y} = \Phi \mathbf{x} \quad (2.10)$$

For 2-D discrete data $x(t_1, t_2), 0 \leq t_1, t_2 \leq N - 1$, $\|\mathbf{x}\|_{TV}$ is defined as $\|\mathbf{x}\|_{TV} = \sum_{t_1, t_2} \sqrt{|D_1 x(t_1, t_2)|^2 + |D_2 x(t_1, t_2)|^2}$, where D is the finite difference $D_1 x(t_1, t_2) = x(t_1, t_2) - x(t_1 - 1, t_2)$ and $D_2 x(t_1, t_2) = x(t_1, t_2) - x(t_1, t_2 - 1)$. TV method is very effective for images that are sparse in the gradient. For example, the following figure illustrates the reconstructions for a popular Logan-Shepp phantom image. As shown in Figure 2.2(d), TV minimization result is exact by just using Fourier samples along 22 radial lines, but the minimum energy reconstruction by setting the unobserved Fourier coefficients to zeros gives much worse result.

Basis Pursuit De-Noising method is a variant of BP method, and is used to deal with noisy sparse signal reconstruction problem (2.8). It can be written as a linearly constrained convex quadratic program, and is considered practically to solve. FOCUSS [99], DS [100] and Lasso method [101-102] are other methods that try to solve a similar problem $\min_{\mathbf{x}} \left(\frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \right)$. For more details see the corresponding reference papers.

Figure 2.2: A simple MRI image reconstruction example: (a) the original Logan-Shepp phantom test image; (b) sampling domain Ω (Fourier coefficients are sampled along 22 approximately radial lines) in the frequency plane; (c) minimum energy reconstruction obtained by setting unobserved Fourier coefficients to zeros; (d) reconstruction obtained by minimizing the total variation (the reconstruction is exact) ([54]).

Because most ℓ_1 -norm convex relaxation methods are based on global optimization, they can get a stable better result in ways that greedy algorithm cannot. However, ℓ_1 -norm methods require fairly heavy computations and are quite slow for large-scale data system. Recently, some fast and large-scale ℓ_1 -norm minimization

algorithms are proposed, like using a Preconditioned Conjugate Gradients (PCG) algorithm to compute the search direction in the interior-point method [113].

2.3.2 Greedy Reconstruction Algorithms

To find out the ideal signal $\mathbf{x}$, we need to determine which columns of Φ participate in the measurement vector $\mathbf{y}$. The greedy algorithms use some iterative processing to expand the support set step by step. At each iteration, the greedy algorithms choose the columns of Φ that are most strongly correlated with the remaining part of $\mathbf{y}$ (that is, the residual of this iteration), and then subtract off this contribution to $\mathbf{y}$ and iterate on the residual. It can be expected, after K iterations, the algorithms will have identified the ideal K -sparse signal $\mathbf{x}$.

The greedy algorithms include MP [1], OMP [60, 87], Regularized OMP (ROMP) [90], Stagewise OMP (StOMP) [89], Tree-based OMP (TOMP) [91], Subspace Pursuit (SP) [92] and Compressive Sampling Matching Pursuit (CoSaMP) [93], and more. In these methods, the measurement matrix Φ is considered as a dictionary with each column vector φ_i of Φ as an atom.

The basic greedy algorithm is MP [1]. It first finds the atom that has the highest correlation with the measurement signal $\mathbf{y}$ and subtracts off the correlated part from the measurement signal, and then iterates this procedure on the resulting residual signal. OMP method adds a least-squares minimization step to each iteration of MP method to obtain the best approximation over the atoms that have already been chosen. This revision significantly improves the behavior of MP algorithm.

OMP method follows the steps below:

1. Initialize $\mathbf{r}^0 = \mathbf{y}$, $\mathbf{x}^0 = \mathbf{0}$, $\Lambda = \mathbf{0}$
2. Find $i = \arg \max_j |\langle \mathbf{r}^{n-1}, \varphi_j \rangle|$
3. Update $\Lambda = \Lambda \cup \{i\}$, $\mathbf{x}^n = \Phi_\Lambda^+ \mathbf{y}$
4. Residual $\mathbf{r}^n = \mathbf{y} - \Phi_\Lambda \mathbf{x}^n$
5. Iterate from step 2 until stopping criterion is fulfilled.

Here Φ_Λ^+ is the pseudo-inverse of the sub-dictionary Φ_Λ . Note that OMP method never selects the same atom twice because the residual is orthogonal to the atoms that have already been chosen.

Different from MP and OMP, StOMP [89] takes a fixed number of stages (e.g. 10), while OMP can take many (e.g. K). StOMP runs much faster than BP and OMP, and so is attractive for solving large-scale problems. The selection step of StOMP method follows a matched filtering and hard thresholding approach to choose a subset of atoms at each time. Figure 2.3 shows the processing of StOMP method.

Figure 2.3: Schematic diagram of StOMP algorithm ([89]).

Tree-based OMP [91] exploits the tree-structured sparse representation as additional prior information for linear inverse problems with limited number of measurements. Tree-based OMP is suitable for the compressible signal that has a sparse representation on the wavelet basis since it should use the tree-structured property. In Tree-based OMP, if a candidate set Σ is selected, all its ancestors are joined to this set. ROMP method applies a preliminary selection and a final selection to carefully choose the atoms that should be added into the estimated support set Λ . SP and CoSaMP methods use a different two-step selection technique to choose the added atoms at each iteration, the size of chosen atoms at each time is fixed (equal to the signal's sparsity level). Better performance is expected for these two-step methods.

Although BP is a more sophisticated approach that can often offer better performance than most greedy algorithms, greedy algorithms received significant attention recently because of their low computational cost and simple geometric interpretation.

2.3.3 Iterative Thresholding Reconstruction Methods

Iterative thresholding algorithms were proposed to handle a class of convex unconstrained optimization problems arising in image restoration and other linear inverse problems [62-63, 108-110]. These problems result from combining a linear observation model with a non-quadratic regularizer (e.g., total variation, or wavelet-based regularization). Thresholding is a simple technique for signal processing.

To solve the sparse regularized optimization problem,

$$\min_{\mathbf{x}} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_0 \quad (2.11)$$

some authors have proposed the Iterative Soft Thresholding (IST) algorithms to approximate the solution [62, 97]. Other authors investigate the properties of Iterative Hard Thresholding (IHT) schemes [63, 108-109, 114]. In particular, [114] shows IHT's convergence to a local minimum under the assumption that $\|\Phi\|_2 < 1$. The convergence proof of the algorithm relied on the fact that after a finite number of iterations, the algorithm does not change the selected subset anymore.

The iterative thresholding method follows the following iterative step:

$$\mathbf{x}^{n+1} = T_\lambda(\mathbf{x}^n + \Phi^H(\Phi\Phi^H)^{-1}(\mathbf{y} - \Phi\mathbf{x}^n)) \quad (2.12)$$

When Φ is an orthonormal matrix, $\Phi\Phi^H = \mathbf{I}_M$, it can be simplified:

$$\mathbf{x}^{n+1} = T_\lambda(\mathbf{x}^n + \Phi^H(\mathbf{y} - \Phi\mathbf{x}^n)) \quad (2.13)$$

where T_λ is a thresholding operator, it denotes hard thresholding operator H_λ or soft thresholding operator S_λ :

$$H_\lambda(x) = \begin{cases} 0, & |x| \leq \lambda \\ x, & |x| > \lambda \end{cases} \quad S_\lambda(x) = \begin{cases} x - \lambda & x \geq \lambda \\ 0 & |x| < \lambda \\ x + \lambda & x \leq -\lambda \end{cases} \quad (2.14)$$

Here λ is the setting thresholding value.

Otherwise, [81] talked about some reweighted iterative algorithm such as Iteratively-Reweighted-Least-Squares (IRLS) for sparse signal reconstruction and it also provides some theoretical analysis about each sparse signal reconstruction algorithm. More details can be seen in [81] and other corresponding references.

Iterative thresholding algorithms are simple and fast. However, most of the algorithms are only guaranteed to find a local minimum which may differ for different initial points. Hence good initialization is needed for better performance. For

example, we can use the result of MP method to initialize IHT algorithm [63]. The algorithms can be used to improve the results calculated with other methods such as MP, and often also discover better set of elements to describe the given signal.

2.3.4 Performance Comparison of CS Reconstruction Methods

This section mainly performs some simple experiments to show several basic algorithm's actual performance, such as signal approximation performance, number of measurements required to identify a sparse signal with a fixed error rate, and the performance of application to 2-D compressible natural images.

We first consider the 1-D sparse signal case and describe the experimental setup. In each trial, we first generate a K -sparse signal $\mathbf{x}$ by randomly choosing K components (out of N) and setting them equal to ± 1 . Then we draw an $M \times N$ Gaussian measurement matrix Φ (the columns are drawn independently at random from Gaussian distribution with mean zero and standard deviation one and then normalized) and perform each recovery method with the data vector $\mathbf{y} = \Phi\mathbf{x}$. Finally, we check if the reconstructed signal is identical with the original signal $\mathbf{x}$. The default convergent condition is set as the error tolerance being smaller than threshold values (default 10^{-6}) or the maximum number of iterations being reached. For each triple (K, M, N) , we perform 100 independent trials. Here we use BP, MP, OMP, IHT and StOMP methods to do the experiments and compare the results.

Figure 2.4 shows Mean Square Error (MSE) between the reconstructed signal and the original one in dimensions $N = 256$ with sparsity level $K = 20$ as a function of measurement numbers by using the above five algorithms. As expected,

when the number of measurements increases, MSE is smaller and the reconstructed signal is closer to the original one. Furthermore, MP, OMP, StOMP and IHT algorithms need more measurements than BP to get the same reconstruction errors. StOMP result is better than MP and OMP. When $M < 80$, OMP method is worse than MP and the reason is that the default MP code uses more iterations to choose more atoms to minimize MSE. However, when the number of measurements is big enough, OMP method can improve the performance and get a better approximation than MP algorithm. The experiment also proves IHT method is just a local optimal method, requiring good initial points. We use MP result to initialize IHT algorithm, the result is slightly better than MP result when samples are big enough; however, if we use OMP result to initialize IHT algorithm, the result is very close to OMP result. However, IHT method can't improve BP results since BP is a global optimization method and it can guarantee the optimal results at the given conditions.

Now we want to know how many measurements are required to identify a sparse signal with a fixed error rate, such as $\text{MSE} < 10^{-8}$. Figure 2.5 presents the measurements needed to recover signals accurately as a function of the sparsity level for the five methods. It shows that for a given sparsity level, BP method requires the fewest measurements to recover correctly, and MP method requires the most measurements to recovery correctly (IHT method is initialized with MP result). With the non-zero number increasing, StOMP and OMP needs much fewer measurements than MP to reconstruction correctly, and their results are closer to BP result. From this figure, it also shows the slopes of lines are nearly linear to $K \log N$, and these results are consistent with the theoretical result [54-57].

Figure 2.4: Mean Square Error (MSE) between the reconstructed signal and the original one (binary sparse signal) as a function of measurement numbers by using different reconstruction algorithms (IHT method is initialized by MP result).

Figure 2.5: The needed measurements to recover signals correctly ($MSE < 10^{-8}$) as a function of the sparsity level for different methods (IHT method is initialized by MP result).

In the following, we also consider the CS reconstruction performance on recovery of a 2-D compressible signal by using the above five methods. We take a part of the image Lena (size 128×128) as an example. We use the Multi-CS scheme [57] to expand the image on wavelet domain (the coarse scale is $j_0 = 4$, the fine scale is $j_0 = 6$), and then the different scales of wavelet coefficients are segregated and sampled with different Gaussian random matrices. The reconstructed results are given in Figure 2.6. For the original image, there are 16384 image pixels. From the experiments, we observe BP reconstruction can get the best reconstructed image with $\text{PSNR}_{\text{BP}} = 40.25\text{dB}$ by using 8756 random measurements; the other methods obtain results $\text{PSNR}_{\text{MP}} = 38.66\text{dB}$, $\text{PSNR}_{\text{MP+IHT}} = 38.05\text{dB}$, $\text{PSNR}_{\text{OMP}} = 37.56\text{dB}$ and $\text{PSNR}_{\text{StOMP}} = 38.79\text{dB}$ by using 8756 random measurements, respectively. This experiment shows that CS approach is also suitable for most natural images since these images are compressible in some basis. Furthermore, although the recovered image from the best 1256-term wavelet coefficients has $\text{PSNR}_K = 40.38\text{dB}$ (slightly better than BP reconstruction), its detailed feature is not as clear as the results from CS reconstructions, such as BP reconstruction result.

Figure 2.6: 128×128 image reconstruction results from 8756 measurements: (a) the original image: a part of 'Lena', (b) recovered from 1256-term best wavelet approximation PSNR=40.386dB, (c) recovered by BP method, PSNR=40.250dB, (d) recovered by MP method, PSNR=38.661dB, (e) recovered by MP +IHT method, PSNR=38.048dB, (f) recovered by OMP method, PSNR=37.557dB, (g) recovered by StOMP method, PSNR=38.785dB.

From all the above experiments, when the number of measurement is enough, all the three kinds of sparse reconstruction algorithms could accurately recover the original signal. Since BP method is a more sophisticated approach based on global optimization, it can get a stable and better reconstruction than that from greedy algorithms and iterative thresholding methods, such as MP, OMP, StOMP and IHT. BP method can also be used with noisy data by solving an optimization problem trading off a quadratic misfit measure with a ℓ_1 -norm of coefficients. Some reference papers show that it can stably suppress noise while preserving a structure that is well-expressed in the dictionary under consideration. Although BP reconstruction offers the best performance, it requires fairly heavy computations and is quite slow for large-scale data system, such as image or video sequence. OMP reconstruction,

especially StOMP reconstruction, is often relatively fast and thus could be used extensively in many applications. The iterative thresholding method is simple and fast, but does not have a unique solution for different initializations; it is therefore important to specify conditions on the fixed points of the algorithm. Considering the trade-off between reconstruction performance and computational cost, StOMP method shows the best performance in our experiments. Every method has some advantage and disadvantage, and therefore one could choose different methods according to different applications' requirements.

2.4 Applications of Compressive Sensing

Compressive sensing addresses transform-based compression schemes' inefficiencies by directly acquiring a compressed signal representation without going through the intermediate stage of acquiring N samples for a length- N signal. Since it emerged, it has been used in a range of different fields, extending from faster data acquisition to higher effective sampling rates and lower communication burden. Particular applications include compressive imaging [64-65], medical imaging [66-68], ADC [69-71], Geophysical Data Analysis [72], compressive Radar [73-75], and communications [76-78].

One example is MRI in medicine [66-68], in which only a limited number of measurements about an object of interest are made, but CS scheme can be adapted to address this kind of questions effectively, see Figure 2.2. As another example of effective CS application, a new digital image/video camera that directly acquires random projections of the signal without first collecting the pixels is proposed in [64-

65]. As shown in the following Figure 2.7, the camera architecture employs a digital micromirror array to perform optical calculations of linear projections of an image onto pseudo-random binary patterns. Its hallmarks include the ability to obtain an image with a single detection element while measuring the image/video fewer times than the number of pixels — this can significantly reduce the computation required for video acquisition/encoding since it completes all these things in a single step. Because the system relies on a single photon detector, it can also be adapted to image at wavelengths that are currently impossible with conventional CCD and CMOS imagers. In specific applications CS principle might enable to dramatically reduce measurement time, dramatically reduce sampling rates, or reduce use of analog-to-digital converter resources.

Figure 2.7: A Compressive Imaging (CI) camera block diagram. Incident light field (corresponding to the underlying image $\mathbf{x}$) is reflected off a Digital Micromirror Device (DMD) array whose mirror orientations are modulated in the pseudo-random pattern supplied by Random Number Generators (RNG), each different mirror pattern ϕ_i produces a voltage at the single photodiode that corresponds to one measurement y_i ([65]).

Chapter 3

Optimized Measurement Matrix for Compressive Sensing

3.1 Introduction

The Shannon sampling theory models signals as band-limited and encodes them through their time-domain samples. It states that to avoid losing information when capturing a signal, the sampling rate must be at least two times the maximum frequency of interest. However, the recently emerged CS method offers an alternative to the classical Shannon theory for sampling signals. It replaces the band-limited signal model by the assumption that the signal is sparse or compressible with respect to some basis or dictionary of wave forms. Based on this assumption, CS offers a joint sensing and compression framework for signal processing [54-56]. Compressive sensing aims to measure the sparse and compressible signals at close to their intrinsic information rate rather than their Nyquist rate [56]. It is based on the revelation that a small group of non-adaptive linear projections of a sparse signal contains enough information for perfect reconstruction. However, rather than directly measuring the original signal, CS framework measures a signal that is a linear combination of the components of the original signal. By employing these non-adaptive linear projections,

the CS measurements preserve the structure of the original signal, and then the original signal can be reconstructed effectively from these measurements by using some ℓ_1 -norm optimization process or greedy algorithms.

One fundamental problem in CS is how to construct a measurement matrix Φ so that the number of measurements is near minimal. Candès and Tao [79] formulated a sufficient condition called RIP Property, which guarantees the performance of a CS measurement matrix. Most work on CS assumes that Φ is drawn at random from some probability distribution, which simplifies the theoretical analysis, and also facilitates a simple implementation [54-57, 79]. Recently, there are some deterministic constructions of the sampling measurement matrices that also satisfy the RIP property [115-117]. However, their guaranteed RIP performance is not comparable to that of the random matrices, and some existing deterministic matrices have certain restrictions on the size of measurement matrices. Hence in this chapter, the work is still based on the random measurement matrix. The proposed method is a modified version of the optimized method proposed by Elad [80]. Elad's method is performed by iteratively reducing the average mutual coherence of the given CS random measurement matrix to get a better-performance sampling matrix. In our proposed method, a simple polynomial function is used as the shrink operation, and a detector is also used to estimate the convergence condition to early stop the iterations. Through these modifications, experimental results show the proposed method can give a better CS reconstruction performance with less optimization time compared to the method given in [80].

The remainder of this chapter is organized as follows: In Section 3.2, the concepts of RIP property and MC are briefly reviewed. The proposed optimization method is presented in Section 3.3. In Section 3.4, the experimental results and reconstruction performance are simulated and discussed, and then an extension to an application for DCS case is also discussed. Finally, we give the conclusions in Section 3.5.

3.2 Measurement Matrix and its Properties

A CS measurement matrix can work based on one important assumption that is referred as Restricted Isometry Property [79]. The RIP property given below is an important sufficient condition to provide theoretical guarantee for exact sparse signal reconstruction.

Definition 1 (RIP [79]): let Ω denote the set of all length- N vectors with at most K non-zero coefficients. An $M \times N$ measurement matrix Φ has the RIP property of order K with restricted isometry constant $\delta \in (0,1)$ if it satisfies

$$(1 - \delta)\|\mathbf{x}\|^2 \leq \|\Phi\mathbf{x}\|^2 \leq (1 + \delta)\|\mathbf{x}\|^2 \quad (3.1)$$

for all K -sparse vectors $\mathbf{x}$. Note that the RIP property implies that for all $K \times N$ submatrices of Φ , the eigen-values of their Gram matrices $\mathbf{G}_K := \Phi_K^T \times \Phi_K$ lay in the interval of $[1 - \delta, 1 + \delta]$. Here Φ_K^T stands for the transpose of sub-dictionary Φ_K . The RIP property guarantees upper bounds for the performance of CS matrices. Another important definition related with the measurement matrix performance is MC of the measurement sampling matrix Φ , defined in [103].

Definition 2 (MC [103]): for a measurement matrix Φ , its mutual coherence is defined as the largest absolute and normalized inner product between different columns in Φ , that is,

$$\mu\{\Phi\} = \max_{1 \leq i, j \leq N, \text{ and } i \neq j} \frac{|\varphi_i^* \varphi_j|}{\|\varphi_i\| \|\varphi_j\|} \quad (3.2)$$

The MC $\mu\{\Phi\}$ provides a measure of the worst similarity between the dictionary's columns. If any two columns have a high coherence, such two closely related columns may confuse any pursuit technique and make the reconstruction fail.

A different way to understand MC is by considering the Gram matrix $\mathbf{G} := \Phi^T \Phi$, computed by using the measurement sampling dictionary Φ after normalizing its each column. The off-diagonal entries in $\mathbf{G}$ are the inner products that appear in equation (3.2). The MC $\mu\{\Phi\}$ is the off-diagonal entry g_{ij} with the largest magnitude. The diagonal entries of $\mathbf{G}$ are all one due to the column normalization. Similarly, the Gram matrix $\mathbf{G}$ can also help to understand the RIP property. If the sum δ_K of any K off-diagonal entries in each column or row is small enough (not very close to 1), then the RIP property of order K will be satisfied by this matrix Φ . It seems that the lower the coherence between each pair of columns of Φ , the higher is the probability that the measurement matrix satisfies the RIP property.

If the measurement matrix Φ is designed such that $\mu\{\Phi\}$ is as small as possible, this allows a wider set of candidate signals to reside under the umbrella of successful CS behavior. While this conclusion is true from a worst-case stand-point, it turns out that the MC as defined above does not do justice to the actual behavior of CS sparse representations and the pursuit reconstruction's performance for a specific

CS measurement matrix. Thus, instead of considering the worst-case performance, an average measure of the coherence is more likely to describe its true behavior [80].

3.3 Optimization of CS Measurement Matrix

3.3.1 Existing Optimization Method

In [80], the author defined a different mutual coherence, which reflects the average behavior of the measurement matrix, given as follows:

Definition 3 ([80]): for a given measurement matrix Φ , its t -averaged mutual coherence is defined as the average of all absolute and normalized inner products between different columns in Φ (denoted as g_{ij} , from the Gram matrix $\mathbf{G} := \Phi^T \Phi$) that are above t . That is,

$$\mu_t\{\Phi\} = \frac{\sum_{1 \leq i, j \leq N, \text{ and } i \neq j (|g_{ij}| \geq t)} |g_{ij}|}{\sum_{1 \leq i, j \leq N, \text{ and } i \neq j (|g_{ij}| \geq t)} 1} \quad (3.3)$$

It is easily seen that as the value of t increases, $\mu_t\{\Phi\}$ grows and approaches $\mu\{\Phi\}$ from below. Also it is obvious from the definition that $\mu_t\{\Phi\} \geq t$. In the optimization process, the author targets this value and minimizes it iteratively through a shrink operation. Otherwise, the author also proposed a slightly different mode $\mu_{t\%}\{\Phi\}$, which targets the average mutual coherence of the top $t\%$ of the inner products.

As the main objective of this method is the reduction of the inner products that are above t in absolute value, firstly the Gram matrix $\mathbf{G}$ of the initial measurement matrix Φ_0 is computed, and then each entry of $\mathbf{G}$ is down scaled through a shrink

operation. The shrink operation causes the resulting Gram matrix to become full-rank in the general case. Thus, it should use Singular Value Decomposition (SVD) by forcing a rank M and then finding a new matrix Φ_{n+1} that could best describe the squared-root $\mathbf{S}_{n+1}$ of the obtained rank- M Gram matrix. Hence, the method in [80] firstly addresses the objective to reduce the value $\mu_t\{\Phi\}$, and then is responsible for the feasibility of the newly generated Gram matrix and the identity of the new measurement matrix with better property. The optimization processing steps of this method is given in Table 3.1. More details are given in [80].

The above algorithm can work for both the exact sparse signal case and the compressible signal case. Actually, for a compressible signal, it can replace Φ_0 with $\Phi_0\mathbf{D}$ (assume the signal is sparse in a basis or dictionary $\mathbf{D}$), and then do the same procedure to get the squared-root $\mathbf{S}_{n+1}$, finally generate Φ_{n+1} by finding that minimizes $\|\mathbf{S}_{n+1} - \Phi\mathbf{D}\|$. Thus it also works for compressible signals. Without loss of generality, the following experiments are restricted to the exact sparse signal case.

3.3.2 The Proposed Optimization Method

In this section, some modifications are made on the above existing method to make it more efficient in applications. Firstly, in the simulations, we observe the coherence of the optimized matrix is different by using different threshold t , which makes the reconstruction performance also vary widely. It will be hard to find the optimal threshold t to get the best optimized measurement matrix for a given initial random matrix Φ . Secondly, since different threshold t and shrink value γ will lead to different convergent speeds, it is natural to adaptively decide the number of iteration.

So we defined an average mutual coherence $\mu_a\{\Phi\}$, similar to the t -averaged MC, given as follows:

Definition 4: for a given matrix Φ , its average mutual coherence $\mu_a\{\Phi\}$ is defined as the average of all absolute and normalized inner products between different columns in Φ (denoted as g_{ij} , from the Gram matrix $\mathbf{G} := \Phi^T \Phi$), that is,

$$\mu_a\{\Phi\} = \frac{\sum_{1 \leq i, j \leq N, \text{ and } i \neq j} |g_{ij}|}{M \times N - N} \quad (3.4)$$

As a modification of the algorithm in [80], the change of this average mutual coherence $\mu_a\{\Phi\}$ between two consecutive iterations is used as a detector to early stop the iteration when the change is smaller than some specific convergence threshold ε . This modification avoids guessing the optimal threshold t for a given matrix. Furthermore, a simple polynomial function is also used to replace the shrink function given in [80], avoiding separate processing of different inner products g_{ij} according to their relations with t .

The proposed shrink operation is given in the following function

$$y = \text{sign}(x) \cdot (|x| - 0.5 \cdot x^2) \quad (3.5)$$

This polynomial function is described graphically in Figure 3.1. As can be seen, the proposed function can easily reduce the variable range of the input values, and make the input values' range $[-1, 1]$ down scaled into range $[-0.5, 0.5]$. Hence, if the entries of Gram matrix $\mathbf{G}$ are used as an input, it can effectively reduce the relatively-large inner products of the given measurement matrix Φ . Besides, compared with the shrink operation in [80], the proposed shrink operation is much simpler and easier to implement. The optimization processing detail is given in Algorithm 2.

Table 3.1: The optimization method proposed by Elad [80]

Algorithm 1: optimization of CS Measurement matrix proposed by Elad [80]

Input: t or $t\%$ -coherence (fixed or relative) threshold,

M -number of measurements,

$Iter$ -number of maximum iterations, and

γ -down-scaling factor

Initialization: set $\Phi_0 \in \mathbb{R}^{M \times N}$ to be an arbitrary random matrix, drawn randomly from the standard i.i.d. Gaussian distribution $N(0, 1)$.

Loop: set $n = 0$ and repeat $Iter$ times:

1) Normalize: Normalize the columns in the matrix Φ_n

2) Compute Gram matrix $\mathbf{G} = \Phi_n^T \Phi_n$

3) Set Threshold: if it's the fixed threshold mode, use t as threshold; if it's the relative threshold mode, choose t such that the average of the top $t\%$ of the off-diagonal entries in current $\mathbf{G}$.

4) Shrink: update the Gram matrix and obtain $\hat{\mathbf{G}}$ by

$$\hat{g}_{ij} = \begin{cases} \gamma g_{ij} & |g_{ij}| \geq t \\ \gamma t \cdot \text{sign}(g_{ij}) & \gamma t \leq |g_{ij}| < t \\ g_{ij} & |g_{ij}| < \gamma t \end{cases}$$

5) Reduce rank: apply SVD decomposition and force the rank of $\hat{\mathbf{G}}$ to be equal to M .

6) Squared-root: build the squared-root of $\hat{\mathbf{G}}$, $\mathbf{S}_{n+1}^T \mathbf{S}_{n+1} = \hat{\mathbf{G}}$, where $\mathbf{S}_{n+1}$ is of size $M \times N$.

7) Choose Φ as the newly generated matrix with better property : $\Phi_{n+1} = \mathbf{S}_{n+1}$

8) Advance: Set $n = n + 1$

End loop

Output: the output of the above algorithm is Φ_{n+1} .

To illustrate the behavior of the above algorithms, we provide an example in Figure 3.2. Considering a random measurement matrix of size 30×200 , whose entries are drawn from i.i.d. zero mean and unit variance Gaussian distribution, we seek the optimized measurement matrices by using Algorithm 1 (fixed and relative threshold modes) and the proposed Algorithm 2. Figure 3.2 presents the histograms of the absolute off-diagonal entries of $\mathbf{G}$ before the optimization and after the optimization with the three methods. As can be seen, all the methods make a shift towards the origin of the histogram after optimization. The proposed optimization method makes the average mutual coherence closer to the origin, and the maximum inner product is just 0.35. That means after optimization, the big off-diagonal entries of $\mathbf{G}$ become smaller and the average inner product of Φ was gradually reduced. Due to the modifications, the proposed method gives a much better result than both fixed and relative threshold modes of Algorithm 1, and it also needs much less time to get convergence.

Figure 3.1: The shrink operation used in the proposed method.

Table 3.2: The proposed method for optimizing CS measurement matrix

Algorithm 2: optimization of CS Measurement matrix

Input: M -number of measurements,

$Iter$ -number of maximum iterations, and

ε -convergence threshold,

Initialization: set $\Phi_0 \in \mathbb{R}^{M \times N}$ to be an arbitrary random matrix, drawn randomly from the standard i.i.d. Gaussian distribution $N(0, 1)$, $\Phi_1 = \Phi_0$ and $(\mu_a\{\Phi\})^0 = 0$.

Loop: set $n = 1$ and repeat till convergence:

1) Normalize: Normalize the columns in the matrix Φ_n

2) Compute Gram matrix $\mathbf{G}^n = \Phi_n^T \Phi_n$

3) Set Threshold: compute $(\mu_a\{\Phi\})^n$ as threshold

4) Shrink: update the Gram matrix and obtain $\mathbf{G}^{n+1}$ by

$$g_{ij}^{n+1} = \text{sign}(g_{ij}^n) \cdot (|g_{ij}^n| - 0.5 \cdot (g_{ij}^n)^2)$$

5) Reduce rank: apply SVD decomposition and force the rank of $\mathbf{G}^{n+1}$ to be equal to M .

6) Squared-root: build the squared-root of $\mathbf{G}^{n+1}$, $\mathbf{S}_{n+1}^T \mathbf{S}_{n+1} = \mathbf{G}^{n+1}$, where $\mathbf{S}_{n+1}$ is of size $M \times N$.

7) Choose Φ as the newly generated matrix with better property : $\Phi_{n+1} = \mathbf{S}_{n+1}$

8) If $|(\mu_a\{\Phi\})^n - (\mu_a\{\Phi\})^{n-1}| \leq \varepsilon$ or if $n = Iter$ iterations, stop the loop and output Φ_{n+1} ; otherwise, set $n = n + 1$, and then go to step 1.

End loop

Output: the output of the above algorithm is Φ_{n+1} .

(a)

(b)

(c)

(d)

Figure 3.2: The histograms of the absolute off-diagonal entries of G before the optimization and afterwards, (a) the initial random matrix (b) matrix optimized by Algorithm 1 with 1000 iterations and $t=0.2$ (fixed threshold mode), $\gamma = 0.5$ (c) matrix optimized by Algorithm 1 with 1000 iterations and t = average of the top 20% of the off-diagonal entries in G^n (relative threshold mode), $\gamma = 0.5$ (d) matrix optimized by the proposed Algorithm 2 with $\varepsilon = 10^{-8}$.

3.4 Experimental Results and Discussions

In this section, we have done some experiments to show the optimized measurement matrices' performance in the CS setting. The experiments are performed on 1-D exact sparse signals and also on 2-D compressible signals. For 1-D exact sparse signals, there are two ways to analyze the performance of the optimized measurement matrix: one is the average reconstruction errors for CS reconstruction problems, which are compared by using the initial measurement matrix and the optimized matrices obtained from the method in [80] and the proposed method; another performance criterion is the probabilities of successful reconstructions. Besides, the computation times for each optimized method are also provided. Finally, we also consider the applications of the optimized measurement matrix in DCS case. In this case, similar performance criteria are illustrated to show the performance gain over the initial random matrix. For 2-D compressible signals, we show the recovery images with Peak Signal-to-Noise Ratio (PSNR) using initial random matrix, Elad's method and the proposed method respectively, by using OMP as the reconstruction algorithm.

3.4.1 Experiments on Exact Sparse Signal Reconstruction

In this subsection, we do two simulations to show the performance of optimized measurement matrix for 1-D sparse signal. The first one shows the average reconstruction errors and the computation time for optimizing processing; the probabilities of successful reconstructions are shown in the second one.

In Experiment 1, for each number of measurements M , $L = 5000$ sparse signals of length $N = 80$ with $K = 4$ non-zero entries are generated to obtain the average performance. The CS reconstruction performance was tested with varying M in the range $10 - 25$. The average reconstruction error was evaluated as a function of M by using OMP method, and the measurements were sampled with the initial measurement matrix and the optimized measurement matrices, respectively. For each M , the initial $M \times N$ matrix Φ_0 was randomly generated from the standard i.i.d. Gaussian distribution, and then the proposed algorithm and the algorithms in [80] were used to generate the optimized measurement matrices. The optimized matrix Φ of method in [80] was done using up to 1000 iterations by setting $\gamma = 0.5$ and using the best chosen value t . (Here the best means we try different $t = 0.1, 0.2, 0.3, 0.5, 20\%$ to choose the one that give the best performance, and then use the best one to run CS reconstructions, and finally compare it with the proposed method. In the trials, we find when M is small, $t = 0.2$ (fixed threshold mode) seems to give the best result; however, when the number of measurements are big enough, $t = 20\%$ (relative threshold mode) gives the best result. Hence both cases were used to compare with the proposed method.) The proposed algorithm used $\varepsilon = 10^{-8}$ and $Iter = 1000$ as the inputs. The results are shown in Figure 3.3.

In this figure, each point represents an average performance of 5000 times' trials. The reconstruction error is computed as $\|x_i - \hat{x}_i^{\text{OMP}}\|_2 = \sqrt{\sum_{i=1}^N (x_i - \hat{x}_i^{\text{OMP}})^2}$ for each signal $\mathbf{x}$ in each trial. As shown in Figure 3.3 (a), the results of all the optimized matrices improve the CS reconstruction performance compared to the initial

random matrix. The proposed algorithm can give a much better reconstruction quality than the initial matrix for all the tested number of measurements. Furthermore, the proposed method can also give a better reconstruction result than both the optimized methods in [80], especially when the measurements are not big enough. For example, at $M = 10 - 15$, the proposed method gives much better results than the one in [80].

Figure 3.3: (a) the average reconstruction error as a function of the number of measurements M , by using the initial random matrix and the optimized matrices by Algorithm 1($t=0.2$ (fixed threshold mode) and $t=20\%$ (relative threshold mode)) and Algorithm 2 (b) the corresponding computation times for optimization by the different methods.

(a)

(b)

Figure 3.4: (a) the probability of successful reconstruction as a function of the number of measurements M , with the initial random matrix and the optimized matrices by Algorithm 1 ($t=0.2$ (fixed threshold mode) and $t=20\%$ (relative threshold mode)) and Algorithm 2 (b) the corresponding computation times for optimization by the different methods.

Figure 3.3(b) shows the computation time needed by every method to optimize the measurement matrix. Since the proposed method uses a detector to observe the convergence and hence stops the iteration early when necessary, it could be faster than the Algorithm 1. This point is clearly shown in Figure 3.3(b), and it illustrates the proposed method is much faster than Algorithm 1 in most points. Therefore, this experiment demonstrates the proposed optimization method speeds up the original optimized method without losing any performance.

The second experiment shows the probability of successful reconstruction (a reconstruction with a ℓ_2 -norm below some threshold is considered as a successful reconstruction) by using different measurement matrices. The experiment setup was similar to the first one. We still generated $L = 5000$ sparse signals of length $N = 80$ with $K = 4$ non-zero entries in each, and then used the initial random matrix and the three optimized matrices to do the measurement sampling and reconstruction, and finally computed the probability of successful reconstruction. That means we count how many times the reconstructions are with error $\|x_i - \hat{x}_i^{\text{OMP}}\|_2 < 10^{-8}$ out of the 5000 trials for each M . As expected, Figure 3.4 shows the optimized matrix of the proposed Algorithm 2 gives the highest successful reconstruction probabilities for all M among all the tested matrices. It is more than 20% higher in some cases. Moreover, as shown above, it also requires much less time to get the optimized matrix.

Some experiments with large-size signal case and noisy perturbation case, and reconstruction performance by BP method have also been done, and these results show similar improvement as the above ones. From these experiments, it is easy to see the optimized matrix of the proposed algorithm can obtain a performance with lower

reconstruction error and higher probability of successful reconstruction, which performs much better than the initial random measurement matrix and the optimized matrices by algorithm in [80]; moreover, the proposed method requires much lower extra computational cost to obtain the goal, therefore it may be suitable to be used in some CS applications. Hence the potential of the proposed optimization method for 1-D sparse signal processing is clearly demonstrated in this subsection.

3.4.2 Experiments on Distributed CS Case

As an extension to the optimized matrix applications, we consider the performance of the proposed algorithm in a simple distributed compressive sensing case. Assume a number of sensors T measure a signal, and each sensor independently encodes the signal by projecting it onto an incoherent random measurement matrix and then transmits just a few of the resulting measurements to a single center collection point. Then a decoder at the center collection point can reconstruct the signal precisely. Since each sensor collects some measurements about the same signal, combining all the measurements together can give a much better reconstruction. Actually, the DCS setting here is very similar to a Multiple Description Coding (MDC) system, where the collected measurements of each sensor can be seen as a description, the decoder at the center collection point of DCS is similar to the central decoder of MDC system, and the decoder of any single sensor can be seen as a side decoder of the corresponding description. Our proposed optimization method can also be used in these cases to give better reconstruction results.

Here, let us see a simple example. Firstly, as the above experiments, we generate $L = 5000$ sparse signals of length $N = 200$ with $K = 8$ non-zero entries in each, and then use $T = 2$ sensors to measure each signal. Finally, we collect $M = 15 - 34$ measurements at each sensor by using an initial random matrix and the optimized matrix of the proposed algorithm, and then in the center collection point we use OMP method to reconstruct the results, which are given in Figure 3.5. From this figure, as expected, DCS reconstruction is much better than the reconstructions of any single sensor, and DCS reconstruction with the optimized matrix of our proposed method can further give a better reconstruction performance than DCS reconstruction with initial random measurement matrix. Note that this case is different from the CS reconstruction with a single encoder and decoder of the previous section, since measurement matrix of each sensor is generated and optimized independently. Because the initial random matrix of each sensor is separately optimized by the proposed algorithm, so the performance gain in DCS reconstruction case is not as much as that in the above CS reconstruction case, which just uses one initial random matrix and then optimizes it as a whole. But it still proves that our proposed optimization method can separately optimize the measurement matrices at each sensor to generate a better reconstruction performance in the DCS framework.

(a)

(b)

Figure 3.5: (a) the average reconstruction error as a function of the number of measurements M in DCS case (b) the probability of successful reconstruction as a function of the number of measurements M in DCS case.

3.4.3 Experiments on 2-D Compressible Signal Reconstruction

In this subsection, the final experiment compared the performance on compressible image using initial random matrix, Elad's method, and the proposed method. Here we simply used OMP as the reconstruction algorithm. A testing image "Cameraman" was used in our experiment. The results are shown in Figure 3.6.

For simplification, the testing image was firstly partitioned into $N = 16 \times 16$ blocks; for each block, we performed 2-D DCT transform and quantized it with quantization value 16 to obtain a K -sparse coefficient representation, and then used an initial random matrix and optimized matrices by Elad's method and the proposed method to collect the measurements. For each block, the DC coefficient was transmitted directly, and then the zero DC and all the AC coefficients are sampled by the above matrices. Elad's method used $t = 0.2, \gamma = 0.5$ and $Iter = 200$ as the inputs. The proposed algorithm used $\varepsilon = 10^{-4}$ and $Iter = 200$ as the inputs.

At the reconstruction side, for each block, firstly the OMP method was used to reconstruct the K -sparse coefficients, which were then 2-D Inverse-DCT (IDCT) transformed to recover each block. Finally the whole image was reconstructed. Figure 3.6 illustrates the reconstructed images from $M = \min(3K, N)$ measurements for each block. For this image, the total used measurements are 35867.

As can be seen in Figure 3.6, the PSNR value of the best K -term DCT coefficient representation is 37.1907 dB. If CS sampling matrices are used to do sampling and reconstruction for the best K -term DCT image, those methods still can give close reconstruction. While the reconstructions by initial random matrix and the

optimized matrix of Elad algorithm result in performance degradations of 4.8898 dB and 2.5399 dB respectively, the proposed method gives the best performance among all the three sampling matrices with performance degradation of just 0.9703 dB. The figures also show the subjective quality of the image reconstructed by using different sampling matrices. As shown in the figures of reconstruction error per pixel, the proposed method give the least reconstruction error per pixel in all the CS reconstruction figures. This experiment shows the proposed method is also effective for the compressible signal's reconstruction.

(a)Original image

(b) Best K -term representation, PSNR=37.1907 dB

(c) Rec. by initial random matrix, PSNR= 32.3009dB

(d) Rec. error per pixel for (c)

(e) Rec. by Algorithm [80], PSNR= 34.6508dB

(f) Rec. error per pixel for (e)

(g) Rec. by the proposed method, PSNR= 36.2204dB

(h) Rec. error per pixel for (g)

Figure 3.6: Reconstructed image from $M = \min(3K, N)$ measurements: (a) the original image, (b) reconstructed image by DCT with the best K -term coefficients, PSNR=37.1907 dB, (c) reconstructed image by using the initial random matrix, PSNR= 32.3009dB, (d) reconstruction error per pixel for (c), (e) reconstructed image by using Elad's algorithm [80], PSNR = 34.6508dB, (f) reconstruction error per pixel for (e), (g) reconstructed image by using the proposed method, PSNR = 36.2204 dB, (h) Reconstruction error per pixel for (g)

3.5 Summary

Compressive sensing is a new technique that aims at sampling and compressing signals at the same time. A crucial ingredient in the CS framework is the use of linear projection matrix that mixes the signal efficiently. This operation has been traditionally chosen as a random matrix. In this chapter, the optimization algorithm given in [80] is modified to give a simple and better construction for optimizing the initial random measurement matrix. We have shown CS sampling and reconstruction simulations on 1-D exact sparse signal case, distributed CS case, and 2-D compressible signal case. The performance of the optimized measurement matrix is studied in terms of average reconstruction errors, computation times, probabilities of successful reconstructions, and PSNR value. The experimental results demonstrate the proposed algorithm can lead to a much better CS reconstruction performance with lower extra computational complexity.

The proposed matrix optimization method may have a problem when it processes the high-dimensional signals, since the proposed method needs explicitly computing and storing the whole CS matrix in the memory, which may exceed the used software's memory limit. Besides, when the CS sampling matrix is big, the computation of SVD in Elad's method and the proposed method will become very slow. However, in the real-life processing of high-dimensional signals, we can use some techniques to solve this problem. For example, we can divide the whole CS matrix into several sub-matrices (each sub-matrix will include some columns of the original matrix), then optimize each sub-matrix individually (as done in DCS case),

and finally use those optimized sub-matrices to process the high-dimensional signals' reconstruction.

The idea of optimizing the measurement matrix could be further studied in the future. One possible extension to the future work is to develop a theoretical link between the average mutual coherence and CS reconstruction performance, so it can give a better justification for the optimization process. Another possible extension is to consider other more efficient optimization methods to generate better reconstruction performance in some specific applications like distributed CS field, since our example just shows a slightly better reconstruction performance in DCS case.

Chapter 4

Backtracking-based Adaptive OMP Algorithm for CS and Sparse Signal Reconstruction

4.1 Introduction

Finding a sparse representation or sparse approximation of a given signal is a major issue in various research areas such as signal compression, inverse problems, blind source separation, and the recently developed compressive sensing approach. As shown earlier, CS is a novel data acquisition and compression technique that aims to measure sparse and compressible signals at close to their intrinsic information rates rather than their Nyquist rates [54, 56]. It is based on the revelation that a small group of non-adaptive linear projections of a sparse signal contain enough information for reconstruction. However, rather than directly measuring the original signal, CS framework measures a signal that is a linear combination of the components of the original signal. Although the sampling step is linear, the reconstruction from the underdetermined system $\mathbf{y} = \Phi \mathbf{x}$ is nonlinear and the reconstruction of $\mathbf{x}$ from $\mathbf{y}$ is generally ill-posed. However, Candès [54] and Donoho's [56] work states that if $\mathbf{x}$

is sparse enough, then exact recovery is possible and $\mathbf{x}$ can be well recovered from only $M = O(K \log(N/K))$ measurements through some nonlinear convex optimization methods, like ℓ_1 -norm basis pursuit method [54-57] or other sparse reconstruction algorithms.

While ℓ_1 -norm optimization techniques play an important role in designing computationally tractable CS reconstructions, their computational complexity is still high enough, which makes them impractical for most applications. For example, for some large-scale data processing applications, like image or video processing, the ℓ_1 -norm optimization techniques will become very slow even though the sizes of the image or video are not very big. In such cases, the need of faster CS reconstruction algorithms is critically important, even if one has to increase the number of required measurements. Hence some iterative greedy algorithms received significant attention recently because of their low computational cost and simple geometric interpretation. OMP [60, 87], ROMP [90], StOMP [89], SP [92] and CoSaMP [93] methods belong to this kind of greedy algorithms. These algorithms will be briefly described in the next section. Such a greedy algorithm is an iterative reconstruction method, the computational complexity of this algorithm is around $O(KMN)$, which is significantly lower than the complexity $O(M^2 N^{3/2})$ of most ℓ_1 -norm convex optimization methods such as BP method when interior point methods are employed [118], especially for the very sparse cases where K is much smaller than N . The basic idea behind the greedy algorithm is to find the support set of the unknown signal sequentially. At each iteration, one or several coordinates of the signal vector $\mathbf{x}$ are selected from testing the correlation values between each column of the measurement

matrix Φ and the residual vector resulted from removing the projection components of the already chosen columns in the previous iterations.

In this chapter, we proposed a modified OMP method, named BAOMP, for sparse signal reconstruction and approximation. Similar to OMP or StOMP method, it follows a stage by stage estimation of the sparsity level K and the true support set of the target signal. Furthermore, like SP or CoSaMP method, it incorporates a simple backtracking step to detect the previous chosen atoms' reliability and then removes the unreliable atoms at each time. By adding the backtracking step, we could correct some wrong reconstruction cases that OMP method will lead to because OMP method can't remove the wrong atoms chosen at the previous iterations. However, unlike SP and CoSaMP methods that need the sparsity level K to be known as an input, the proposed method doesn't need K as prior information, and it could adaptively estimate the sparsity level according to the underlying signal. The numerical results show that the proposed BAOMP method outperforms all the above mentioned OMP-type methods and it demonstrates a beneficial impact on the sparse signal reconstruction and approximation.

The rest of this chapter is organized as follows: In Section 4.2, several greedy algorithms are briefly introduced and studied, and the detailed description of the proposed Backtracking-based Adaptive OMP method is also presented. The experimental results and the performance gain obtained by the proposed method are simulated and discussed in Section 4.3. Finally, we give the conclusions in Section 4.4.

4.2 Backtracking-based Adaptive Orthogonal Matching Pursuit

In this section, we first present a brief overview of some existing greedy algorithms for sparse signal reconstruction and approximation, and then the proposed BAOMP method is also presented in detail. In these greedy methods, the measurement matrix Φ is considered as a dictionary with each column vector φ_i of Φ as an atom. For the underdetermined sparse reconstruction problem $\mathbf{y} = \Phi \mathbf{x}$, each iteration of a greedy algorithm is composed of two major successive steps: 1) **Atom selection**: find one or several atoms that have the largest correlations with the residual and add them to the estimated support set Λ ; 2) **Update**: update the approximate coefficient $\mathbf{x}_\Lambda$ of the selected coordinates listed in set Λ to minimize the new residual error $\min \|\mathbf{y} - \Phi_\Lambda \mathbf{x}_\Lambda\|$ and then update the residual for the next iteration, as shown in Figure 4.1.

4.2.1 Existing Greedy Algorithms

The basic greedy algorithm is MP method [1]. MP method provides a practical solution by iteratively decomposing a measurement signal into a linear combination of atoms that are selected from a redundant dictionary. It first finds the atom that has the highest correlation with the measurement signal vector $\mathbf{y}$ and subtracts off the correlated part from the measurement signal, and then iterates this procedure on the resulting residual signal. OMP [60, 87] method is a variation of MP method. It adds a least-squares minimization step $\min \|\mathbf{y} - \Phi_\Lambda \mathbf{x}_\Lambda\|_2$ to each iteration of MP method to

obtain the best approximation over the atoms that have already been chosen in the estimated support set Λ . OMP method removes MP method's drawback by optimizing all the approximate coefficients in the estimated support set Λ to obtain the minimum ℓ_2 -norm error with the set of all selected atoms. That is, it updates the coefficients $\mathbf{x}_\Lambda$ of the selected coordinates as $\mathbf{x}_\Lambda = \Phi_\Lambda^+ \mathbf{y}$ replacing the update $\mathbf{x}_\Lambda = \Phi_\Lambda^T \mathbf{y}$ in MP method. Here Φ_Λ^T stands for the transpose of sub-dictionary Φ_Λ , and $\Phi_\Lambda^+ = (\Phi_\Lambda^T \Phi_\Lambda)^{-1} \Phi_\Lambda^T$ is the pseudo-inverse of the sub-dictionary Φ_Λ . This revision significantly improves the behavior of MP algorithm. Figure 4.1 (a) describes the schematic diagram of the iterations in OMP method.

Note that MP and OMP algorithms just choose one atom at a time. Unlike them, some greedy algorithms choose several atoms at each iteration, like StOMP [89], ROMP [90], SP [92] and CoSaMP [93] methods. At each iteration of these algorithms, several coordinates of the signal vector $\mathbf{x}$ (or several atoms of the dictionary Φ) are selected from testing the correlation values between the columns of Φ and the resulting residual vector $\mathbf{r}^{n-1}$ from last iteration. The selection step of StOMP method follows a matched filtering and hard thresholding approach to choose a subset of atoms at each time. While for ROMP method, it applies a preliminary selection and a final selection to carefully choose the atoms that should be added into the estimated support set. The preliminary selection is quite similar to that of StOMP method to choose an atom set C , and then the final selection is set to choose the largest subset Σ of those coordinates whose atoms' correlation values have comparable magnitudes from the preliminarily selected atom set C . Then the estimated support set Λ is updated by adding the new discovered atoms into $\Lambda = \Lambda \cup \Sigma$, and the residual

signal is also updated by subtracting the measurement vector $\mathbf{y}$ from its projection onto the subspace spanned by the atoms in the current estimated support set Λ .

SP and CoSaMP methods use another kind of two-step selection technique to choose the added atoms at each iteration, where the size of chosen atoms at each time is fixed (equal to the sparsity level K of the given signal). Both SP and CoSaMP algorithms follow a very similar approach. At the preliminary selection step of SP method a K -atom ($2K$ atoms selected in CoSaMP method) set $\mathcal{C}^n$, which includes the atoms that have the K largest correlations between the columns of Φ and the residual $\mathbf{r}^{n-1}$, is chosen, and a candidate set $\{\Lambda_{UC}^n\}$ is created by the union of the K atoms in the estimated support set Λ and the current chosen K atoms in $\mathcal{C}^n$ ($2K$ atoms in CoSaMP method). At the final selection step, they first solve a least square solution $\mathbf{x}_{\Lambda_{UC}^n} = \Phi_{\Lambda_{UC}^n}^+ \mathbf{y}$, and then update the estimated support set Λ by choosing the subset of K atoms that are corresponding to the K largest values in magnitude of the least square solution $\mathbf{x}_{\Lambda_{UC}^n}$. The resulting residual signal is finally updated in a way similar to that of other OMP-type methods. Compared with other OMP-type methods, the major innovation of SP and CoSaMP methods is the backtracking technique used in the final selection step, which chooses the K more reliable atoms from the current candidate set $\{\Lambda_{UC}^n\}$, and hence enables the algorithms to delete the unreliable atoms that are added wrongly in the previous iterations. The schematic diagram of the iterations in SP method is shown in Figure 4.1 (b).

While SP and CoSaMP methods improve the performance of OMP-type methods and have a good trade-off between the reconstruction performance and computational complexity, they require the sparsity level K to be known as a prior for

exact signal recovery. However, in most practical CS and other sparse reconstruction problems, this piece of information could not be known before the reconstruction. One way out is to guess the value of K before using SP and CoSaMP methods. [93] provides a simple method to estimate this parameter. However, as shown in [119] (it proposes a modified SP method in the case K is unknown, which is different from our method and performs worse than SP method for binary sparse signal reconstruction case), using an incorrectly estimated value of K , the performance of SP and CoSaMP algorithms will degrade quickly. Hence, we propose a new OMP-type method in this chapter, which does not need the sparsity level as known information.

(a) The schematic diagram of the iterations in OMP

(b) The schematic diagram of the iterations in SP

(c) The schematic diagram of the iterations in BAOMP

Figure 4.1: Schematic diagrams of the iterations in some OMP-type methods: (a) OMP method (b) SP method (c) the proposed BAOMP method.

4.2.2 The Proposed BAOMP Algorithm

In this subsection, the proposed Backtracking-based Adaptive OMP method is presented, and then the property of backtracking scheme and its significance on the reconstruction performance are also described.

The proposed BAOMP method chooses several atoms each time to expand the estimated support set Λ stage by stage; it adaptively chooses several candidate atoms at each iteration and unlike SP/CoSaMP methods, it does not need to know the sparsity level K as prior information. On the other hand, like SP and CoSaMP methods, by using the backtracking strategy the proposed method is likely to remove some atoms chosen wrongly in the previous processing and identify the true support set more accurately. The process of the proposed BAOMP method is shown in the following Table 4.1.

As shown in Table 4.1, Λ is the estimated nonzero-coordinate support set up to the current iteration and matrix Φ_Λ consists of the columns of Φ with indices listed in Λ . Like most of OMP-type methods, the first step of the proposed method is to select the candidate set C^n whose correlations between the columns of Φ and the residual vector $\mathbf{r}^{n-1}$ are not smaller than $\mu_1 \cdot \max_{j \in \Omega} |\langle \mathbf{r}^{n-1}, \phi_j \rangle|$. Here $\Omega = [1, 2, 3, \dots, N]$ is the whole coordinate set, and the constraint $|C^n| \leq M - |\Lambda|$ is used to make sure the inverse matrix of $\Phi_{\Lambda \cup C^n}$ exists. We use a constant μ_1 to adaptively decide how many atoms are chosen at each time. When $\mu_1 = 1$, it means we just choose the maximum-correlation atom at each time, like the standard OMP method. When μ_1 becomes smaller, it will choose more atoms at each time, and sequentially

may speed up the algorithm. Since a final refined selection step follows, this preliminary coarse selection may not affect the reconstruction performance too much. At the final selection step, the proposed method removes some atoms whose approximate coefficients $\mathbf{x}_{\Lambda UC^n}^n$ are smaller than μ_2 times the maximum amplitude of $\mathbf{x}_{C^n}^n$ ($\mathbf{x}_{C^n}^n$ are the entries of $\mathbf{x}_{\Lambda UC^n}^n$ corresponding to the coordinates in the current selected candidate set C^n). Here μ_2 is also a constant to adaptively control the number of atoms that should be deleted at each time. This kind of refined procedure has some intuitive justification. The greedy algorithm's objective is to find the K -sparse representation whose approximate coefficients are the K biggest ones and hence to reduce the approximate error sequentially. In the ideal case, it is expected that the approximate coefficients of the previously chosen atoms are bigger than that of the currently chosen atoms. At some iteration, if the newly chosen atoms make the approximate coefficients of the previously chosen atoms become much smaller, it is more likely that these previously selected atoms are chosen wrongly at the previous iterations. Therefore, the proposed method uses the above backtracking technique to remove the atoms with relatively small approximate coefficients. Since the sparsity level K is not an input parameter for the proposed method, the proposed BAOMP method will stop when the current residual vector's ℓ_2 -norm $\|\mathbf{r}^n\|_2$ is smaller than a certain threshold ε or when the maximum number of iterations is reached. A schematic diagram of the iterations in the proposed method is described in Figure 4.1 (c). For comparison, diagrams of OMP and SP algorithms are also provided in Figure 4.1 (a) and Figure 4.1 (b), respectively.

Table 4.1: The proposed Backtracking-based Adaptive OMP method

BAOMP method
Input: Φ $-M \times N$ measurement sampling matrix
$\mathbf{y}$ -Sampled measurement vector
μ_1 -a preset atom-adding constant threshold in $[0,1]$
μ_2 -a preset atom-deleting constant threshold in $[0,1]$
ε -convergence threshold to stop the iterations
n_{max} -number of maximum iterations allowed
Initialization:
$\mathbf{x}^0 = 0$ (initial solution)
$\mathbf{r}^0 = \mathbf{y}$ (initial residual)
$\Lambda = \emptyset$ (estimated support set, empty at the beginning)
$C^0 = \emptyset$ (candidate set)
$\Gamma^0 = \emptyset$ (delete set)
Loop: set $n = 1$ and repeat till convergence:
1) Preliminary selection: find the candidate set C^n by choosing the atoms with the absolute values of correlations satisfying $ \langle \mathbf{r}^{n-1}, \varphi_{C^n} \rangle \geq \mu_1 \cdot \max_{j \in \Omega} \langle \mathbf{r}^{n-1}, \varphi_j \rangle $, $\Omega = [1, 2, 3, \dots, N]$ and with $ C^n \leq M - \Lambda $;
2) Final selection (Backtracking step): compute $\mathbf{x}_{\Lambda \cup C^n}^n = \Phi_{\Lambda \cup C^n}^+ \cdot \mathbf{y}$, and select atom-deleting set Γ^n that satisfy $ \mathbf{x}_{\Lambda \cup C^n}^n < \mu_2 \cdot \max \mathbf{x}_{C^n}^n $;
3) Update: update $\Lambda = \{\Lambda \cup C^n\} \setminus \Gamma^n$, the coefficients $\mathbf{x}_{\Lambda}^n = \Phi_{\Lambda}^+ \cdot \mathbf{y}$ and the residual $\mathbf{r}^n = \mathbf{y} - \Phi_{\Lambda} \mathbf{x}_{\Lambda}^n$
If $\ \mathbf{r}^n\ _2 < \varepsilon$ or if $n = n_{max}$, quit the iteration; otherwise, set $n = n + 1$, go to step 1 and continue with a new iteration.
End loop
Output: the estimated support set Λ and the non-zero values $\mathbf{x}_{\Lambda}^n = \Phi_{\Lambda}^+ \cdot \mathbf{y}$.

The most important difference between OMP method and the proposed BAOMP method is that the proposed method adds a backtracking step to generate the estimate Λ of the true support set. In most of OMP-type methods, during each iteration the algorithms select one or several coordinates that represent good partial support set estimate and adds them to the estimated support set Λ . Once an atom is added to Λ , it always belongs to the estimated support set throughout the remainder of the whole process. However, once a wrong coordinate has been chosen and added to Λ , it will make the remainder process totally incorrect. Especially when the number of measurements is not big enough, with high probability OMP method chooses a wrong atom at some iterations. To correct some wrong reconstruction cases of OMP method, strict selection rules are needed to ensure that the added atoms belong to the true support set. Thanks to the backtracking technique of BAOMP, the proposed method provides the double checks of the chosen atoms' reliability. That is, the chosen atoms are firstly checked by considering the correlation with the residual, and finally checked by observing the approximate coefficients on the union subspace of the estimated support set Λ and the candidate set C^n . This kind of double-check procedure could remove most unreliable atoms in the current candidate set C^n ; furthermore, a coordinate in Λ , which is considered reliable in some iteration but shown to be wrong at a later iteration, can be removed from the estimated support set Λ freely. This backtracking refinement of the estimated support set Λ method will, as shown later, lead to much better sparse reconstruction and approximation performance.

Let us see an example to illustrate the above point clearly. Figure 4.2 shows the reconstruction results of a $K = 4$ sparse signal of length $N = 40$ with $M = 10$

measurements by using OMP method (used the default setting of OMP in the software toolbox ‘SparseLab’, available online [120]) and the proposed BAOMP method ($\mu_1 = 0.6$, $\mu_2 = 0.6$, $n_{\max} = 10$, $\varepsilon = 10^{-6}$ as the input parameters).

Figure 4.2 (a) shows our proposed BAOMP method can give a perfect reconstruction but OMP method cannot. If we observe the atom-picking processing of OMP method, we will understand how badly the result will be if some wrong atoms are picked at the beginning. The original signal’s non-zero coordinates are {39 13 12 33}, but OMP sequentially chooses {4 13 39 23 12 22 35 38 27 30} as the non-zero coordinates. Because the first chosen atom is already wrong, so it is hard to pick out the correct coordinates in the following iterations. However, in the proposed method, as shown in Figure 4.2 (b), the first iteration chooses the preliminary candidate set $C^1 = \{4\ 33\ 16\ 13\ 14\ 10\ 21\ 29\ 19\}$ (marked as red bars in the first iteration). Then at the final selection step, atoms in the set $\Gamma^1 = \{10\ 14\ 16\ 29\}$ (the red bars under the blue line in the first iteration of Figure 4.2 (b)) are removed since the approximate coefficients of these coordinates are smaller than $\mu_2 \cdot \max(|\mathbf{x}_{C^1}^1|)$ (marked as the blue line). So the estimated support set is $\Lambda = \{4\ 33\ 13\ 21\ 19\}$ after the first iteration. At iteration 2, the first chosen atoms are $\{39\ 24\ 29\ 36\ 26\}$, and since the coefficients of atoms in $\Gamma^2 = \{4\ 19\ 21\ 24\ 26\ 29\ 36\}$ are smaller than $\mu_2 \cdot \max(|\mathbf{x}_{C^2}^2|)$, these atoms should be removed. So the atoms in Λ at the end of iteration 2 are $\{13\ 33\ 39\}$. At the third iteration, the preliminarily chosen atoms C^3 are $\{12\ 15\}$. The approximate coefficient of atom 15 is smaller than $\mu_2 \cdot \max(|\mathbf{x}_{C^3}^3|)$ and it should be removed, so the final

estimated atoms in Λ are exactly equal to the true support set $\{33\ 13\ 39\ 12\}$. Hence, the proposed method uses three iterations to give the exact reconstruction.

This example clearly shows that because of the added backtracking step, the proposed BAOMP method improved the reconstruction performance greatly. As shown in Figure 4.2 (b), at the first iteration, the backtracking step of the first iteration removes most of the wrongly chosen atoms, but some wrong atoms are still kept in the estimated support set Λ , like atoms $\{4\ 19\ 21\}$. However, in the second iteration, the approximate coefficients of these atoms become much smaller because of the effect of the newly added atoms, so the wrong atoms $\{4\ 19\ 21\}$ chosen at the first iteration are eventually removed at the second iteration. After removing the wrong atoms, the right atoms will more likely be chosen in the following iterations.

(a) The reconstruction results by OMP and BAOMP

(b) The approximate coefficients of chosen atoms changing with iterations in BAOMP method

Figure 4.2: A sparse signal reconstruction example by using OMP and BAOMP methods.

Note that one important innovation of the proposed method is the use of $\mathbf{x}_{C^n}^n$ in the backtracking step. Here we use the maximum $\max|\mathbf{x}_{C^n}^n|$ of the current chosen

atoms in $\mathcal{C}^n$, rather than the maximum $\max |\mathbf{x}_{\Lambda \cup \mathcal{C}^n}^n|$ of all chosen atoms in $\{\Lambda \cup \mathcal{C}^n\}$, as a benchmark to decide the deleted atoms. As can be seen in the third iteration of Figure 4.2 (b), if we use $\max |\mathbf{x}_{\Lambda \cup \mathcal{C}^n}^n|$ as a benchmark to compare the approximate coefficients, the coordinate $\{12\}$ will never be chosen since the approximate coefficient of coordinate $\{12\}$ is always much smaller than $\mu_2 \cdot \max(|\mathbf{x}_{\Lambda \cup \mathcal{C}^3}^3|)$ (the maximum is the atom $\{13\}$). Hence it is not possible to get the perfect reconstruction. However, since we use $\max |\mathbf{x}_{\mathcal{C}^n}^n|$ as a benchmark and the maximum is the atom $\{12\}$ itself in current candidate set $\mathcal{C}^3$, the proposed method can choose this atom and get the exact reconstruction. It also can be seen that the method always chooses the relatively large-amplitude coordinates first and then chooses the smaller ones. Hence it could adaptively expand the estimated support set Λ stage by stage according to their significance. This is the primary reason why the proposed method does not require prior knowledge of K but is still capable of adaptively estimating it on the run.

Like the proposed BAOMP method, SP and CoSaMP methods also incorporate the idea of backtracking. However, both SP and CoSaMP methods assume that the sparsity level K is known as prior information, even though K may not be available in many sparse reconstruction and approximation applications. But the proposed BAOMP method can achieve the blind sparse signal reconstruction without knowing the sparsity level K as a priori. Besides, we can adaptively choose the numbers of added and deleted atoms at each iteration according to the feature of the given underlying signals, unlike SP and CoSaMP algorithms that fix the numbers of added atoms and deleted atoms equal to K at each time. That is, if K is small, fewer atoms will be chosen or deleted, and if K is large, more atoms will be added or removed.

Furthermore, in the proposed method, at the first several iterations, the numbers of added and deleted atoms may be larger, but after most corrected atoms have been chosen, the chosen atoms will become much smaller, hence it is possible to speed up the convergence. The choice is totally based on the considered signal, not based on some pre-estimated K , therefore it may bring some adaptive property. Finally, by adjusting μ_1 and μ_2 , the proposed method will bring some flexibility between computational complexity and reconstruction performance. These two parameter values may be adaptively chosen according to the particular underlying signals. For example, as iteration progresses, the residual will increasingly be obtainable from fewer non-zero values (atoms) since several correct atoms have already been chosen. Since it is known that reconstruction with a smaller K is easier, later iterations may use larger values of μ_1 and μ_2 . The following numerical results in the following section show our proposed method outperforms other OMP-type methods in extensive simulations.

4.3 Experimental Results and Discussions

In this section, we simulated a series of experiments demonstrating the benefits of the proposed BAOMP method. We compared the simulation results of the proposed BAOMP method with some other greedy algorithms (including OMP, ROMP, StOMP and SP methods; since CoSaMP method is very similar to SP, it is not included here) and the ℓ_1 -norm convex optimization algorithm (BP). BAOMP method is shown to outperform the other greedy algorithms and BP algorithm, especially for Gaussian sparse signal reconstruction and approximation. We also

demonstrate the proposed approach is robust and applicable to a broad array of problems, providing examples of exact sparse signal reconstruction, noise-aware sparse approximation, and 2-D image recovery.

4.3.1 Experiments on 1-D Exact Sparse Signal Reconstruction

We present the reconstruction performance of the proposed method for exact Gaussian or binary sparse signal recovery in this subsection, and compare it with all the above mentioned methods. In next subsection, we also show the average needed iterations of the proposed method for perfect reconstruction of such signals. The effect of different parameters μ_1 and μ_2 is also briefly analyzed through an example.

In the first experiment, the original signals were Gaussian or binary sparse signals with length $N = 256$. To show the probability of the exact reconstruction as a function of the sparsity level K for the given number of measurements $M = 128$, different sparsity levels were chosen from $K = 5$ to $K = 70$ with a step size of 5. For each case, the support set of size K was selected uniformly at random and the non-zero values were generated from a standard Gaussian distribution or a binary distribution. An $M \times N$ measurement matrix Φ was randomly generated from the standard i.i.d. Gaussian distribution in each trial. 500 independent simulations were performed for each K to calculate the probabilities of exact reconstruction by using the above mentioned algorithms. The reconstruction was considered as an exact reconstruction when the maximum magnitude difference between the original signal $\mathbf{x}$ and the reconstruction $\mathbf{x}^{\text{rec}}$ was smaller than 10^{-3} , that is, $|x_i - x_i^{\text{rec}}|_{\max} < 10^{-3}$.

In the experiments, OMP method used K iterations, StOMP and BP methods used the default settings (OMP, StOMP and BP tools used are SparseLab [120]), and ROMP and SP methods used the parameters given in [90] and [92], respectively. Note that exact knowledge of K is assumed for ROMP and SP, without which they would perform much worse. The proposed method used $\mu_1 = 0.4$, $\mu_2 = 0.6$, $n_{\max} = M$, $\varepsilon = 10^{-6}$ as the input parameters. Figure 4.3 (a) and Figure 4.3 (b) show the results of performance for Gaussian and binary zero-one sparse signals, respectively.

As can be seen in Figure 4.3 (a), for Gaussian sparse signals, the performance of the proposed BAOMP method is much better than all other algorithms, even including BP ℓ_1 -norm minimization method. While all the other algorithms start to fail when the sparsity level $K \geq 40$, the proposed method still gives nearly 100% successful reconstruction until $K \geq 50$. The critical sparsity [92], the sparsity level at which the recovery rate drops below 100%, is a very interesting point in sparse signal reconstruction. The proposed method has a higher critical sparsity level $K = 50$. On the other hand, when the original signals are not sparse enough or the measurements are not enough, the proposed BAOMP method still gives a much higher probability of exact reconstructions. For example, at the point $K = 65$, when all the other methods give nearly 0 probabilities of exact reconstructions, the proposed method still gives a more than 0.6 probability of perfect reconstruction. Furthermore, as shown in Figure 4.3 (b), for the binary sparse signal, which is a difficult case for OMP-type methods, all the OMP-type methods are worse than BP ℓ_1 -norm method, but the proposed BAOMP method still gives the best performance among the greedy algorithms.

(a) Gaussian case: probability of exact reconstruction vs. signal sparsity level K

(b) Binary case: probability of exact reconstruction vs. signal sparsity level K

Figure 4.3: Simulations of the exact reconstruction rate vs. the signal sparsity level K by using different methods: (a) Gaussian sparse signal, (b) binary zero-one sparse signal.

In Experiment 2, the Normalized Mean Square Error (NMSE) for various K is observed. For each (K, N, M) , we used the same setup as the first experiment, and take the average MSE over 500 times' trials divided by the input signal's variance as the

NMSE performance. As can be seen in Figure 4.4, the proposed BAOMP method gives the lowest NMSE values nearly for all Gaussian cases. For binary cases, BP gives the best performance when $K \geq 35$, but the proposed method is the best when K is small and gives the best performance among all the considered OMP-type methods.

(a) Gaussian case: NMSE vs. signal sparsity level K

(b) Binary case: NMSE vs. signal sparsity level K

Figure 4.4: Simulations of NMSE vs. the signal sparsity level K by using different methods: (a)

Gaussian sparse signal, (b) binary zero-one sparse signal.

(a) Gaussian case: probability of exact reconstruction vs. number of measurements M

(b) Binary case: probability of exact reconstruction vs. number of measurements M

Figure 4.5: Simulations of the exact reconstruction rate vs. the number of measurements M by using different methods: (a) Gaussian sparse signal, (b) binary zero-one sparse signal.

In Experiment 3, we observe the probability of exact reconstruction as a function of the number of measurements M , given a fixed sparsity level K . We used the same setup as the first experiment, except fixed $K = 30$ and changed M from 60 to 160 with step size of 10. Figure 4.5 demonstrates the experimental results. As can

be seen again, the proposed BAOMP method gives the best reconstruction performance for Gaussian sparse signal. For the binary sparse signal, BP ℓ_1 -norm minimization gives the best performance, but the proposed BAOMP method gives the best reconstruction performance among all the considered OMP-type methods.

4.3.2 Effect of Parameter Choices on Performance

We now show that in spite of its superior performance, BAOMP method requires moderate number of iterations, much smaller than the sparsity level K . Experiment 4 considers the convergence of the proposed BAOMP method, and counts the average number of iterations needed to perfectly reconstruct an arbitrary K -sparse signal. To generate Figure 4.6, we used $N = 256$, $M = 128$, selected different values of K , and ran 500 times' simulations for each case. We used different input parameters μ_1 and μ_2 to do the experiments. Figure 4.6 depicts the average required number of iterations for perfect reconstruction as a function of the sparsity level K . It clearly shows that when the number of measurements is large enough, the needed iterations for perfect reconstruction of a Gaussian or binary K -sparse signal are always smaller than K (the slopes of the average number of iterations vs. K for all cases are much smaller than 1), especially for large K . Although the different choices of parameters μ_1 and μ_2 need different average times of iterations, they all show a similar convergence slope.

(a) Gaussian case: average needed number of iterations vs. signal sparsity level K

(b) Binary case: average needed number of iterations vs. signal sparsity level K

Figure 4.6: Convergence of the proposed method for Gaussian and binary sparse signal when the number of measurements M is fixed.

In Experiment 5, we consider the convergence of the proposed method as a function of the number of measurements M while we fixed $K = 30$. The result is

given in Figure 4.7. As shown in Figure 4.7(a), for Gaussian case, the average needed times of iterations do not change much with different M . (With M increasing, it needs slightly less iterations to convergence. This reduction makes sense since when the sparsity level is fixed, the more measurements are collected, the higher probability it will give for accurate reconstruction, and hence it will be easier to find the correct support set and get convergence.) But the choices of different μ_1 and μ_2 have a big effect on the average needed iterations, such as the reconstruction with $\mu_1 = 0.4$, $\mu_2 = 0.6$ just needing nearly half of the average iterations needed by $\mu_1 = 0.8$, $\mu_2 = 0.6$. Note that this is along expected line since a smaller μ_1 will capture more atoms in each iteration. The binary case also shows a similar convergence result, as can be seen in Figure 4.7(b). Except the cases $M < 135$, the average number of iterations tends to decrease as the measurements increase. This indicates that when the measurements are not enough, more iterations are required to converge. However, the average needed iterations in all the cases are smaller than the sparsity level $K = 30$.

(a) Gaussian case: average needed number of iterations vs. measurements M

(b) Binary case: average needed number of iterations vs. measurements M
Figure 4.7: Convergence of the proposed method when sparsity level K is fixed.

From the above two experiments, it is observed that with a proper choice of parameter μ_1 and μ_2 , the required iterations for perfect reconstruction of a K -sparse signal are much smaller than K . The computational complexity of OMP-type methods depends on the number of iterations needed for perfect reconstruction. In each iteration of the proposed method, the correlation computation requires MN computations, and the computation of the approximate coefficients can be computed with the marginal cost of at most $O(L^2M)$ computations by the Modified Gram-Schmidt algorithm [121]. Here L is the number of the atoms in the set ΛUC^n at the current iteration, which typically is $L = O(K)$, and in the worst case $L = M$. Therefore, the total computations of each iteration are roughly $O(L^2M + MN)$. So the computational complexity of the proposed BAOMP method is at most $O(M(L^2 + N)K)$ for a K -sparse signal reconstruction. When the signal is very sparse, in particular, when $L^2 =$

$O(K^2) \leq O(N)$, the total complexity of BAOMP method is at most $O(MNK)$, similar to OMP method.

The above Experiment 4 and 5 show that the choices of different parameters μ_1 and μ_2 have an effect on the average required iterations. Hence, the following experiment 6 is done to observe the effect on the reconstruction performance by using different parameters μ_1 and μ_2 , whose results are shown in Figure 4.8. As can be seen in Figure 4.8(a) for Gaussian case, with μ_2 fixed to 0.7, the reconstruction performance of $\mu_1 = 0.5$ gives the best result when μ_1 is changed from 0.3 to 0.7; with fixed $\mu_1 = 0.5$, $\mu_2 = 0.7$ gives the best performance. Similarly for the binary case, $\mu_1 = 0.4$, $\mu_2 = 0.6$ shows the best result in our experiments. From all the above three experiments, considered from both the computational complexity (average needed iterations) and the reconstruction performance, $\mu_1 = 0.4$, $\mu_2 = 0.6$ seems a good choice for all considered sparse signal reconstructions.

(a) Gaussian case: probability of exact reconstruction vs. number of measurements M

(b) Binary case: probability of exact reconstruction vs. number of measurements M

Figure 4.8: The effect of different parameter choices on the exact reconstruction rate vs. number of measurements M by using the proposed method: (a) Gaussian sparse signal (b) binary sparse signal.

4.3.3 Experiments on 1-D Noisy Perturbation Reconstruction

All the above experiments consider the exact K -sparse signal reconstruction problems. However, in most applications, the measurement vector $\mathbf{y}$ may be subjected to an additive noise or the sparse signal representation may include some noise. So in the following experiments of this subsection, as in reference [92], we also considered the following two kinds of noise perturbations:

- 1) **Signal perturbation:** the signal elements on the support set Λ are unchanged but the signal entries out of the support set are perturbed by i.i.d. Gaussian noise $N(0, \delta_s^2)$;
- 2) **Measurement perturbation:** the collected measurement vector $\mathbf{y}$ is

perturbed by $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$, $\mathbf{z} \in N(0, \delta_z^2)$.

Firstly, let us see a sparse approximation example in signal perturbation case, with $N = 256$, $K = 15$ and $M = 60$. The entries on the support set Λ were generated from Gaussian distribution $N(0,1)$ and then corrupted by i.i.d. Gaussian white noise $N(0, \delta_s^2 = 10^{-4})$. The results are given in Figure 4.9. The proposed method used $\mu_1 = 0.4$, $\mu_2 = 0.6$, $n_{\max} = M$, $\varepsilon = 8N\delta_s^2$ as the input parameters. All the other methods also used $\varepsilon = 8N\delta_s^2$ as an input to control the convergence, and OMP, ROMP and SP also used K as a priori. As shown in Figure 4.9(h), our proposed method gives the best reconstruction performance with the smallest reconstruction error $\sum(\mathbf{x}_\Lambda - \mathbf{x}_\Lambda^{\text{rec}})^2 / K$ (MSE on the support set Λ). From Figure 4.9(h), it is easy to see that the 14 coordinates with the 14 largest amplitudes in the support set are found correctly by using the proposed BAOMP method, and just the coordinate with the smallest nonzero amplitude is not found. However, all the other methods find some wrong coordinates even though the sparsity level K is known in some methods. In fact, our proposed method does not use the information of the sparsity level K , and it is possible to find all the correct coordinates if K is known in the proposed method.

Figure 4.9: A reconstruction example for signal perturbation case: (a) original sparse signal (b) original signal perturbed by i.i.d. Gaussian noise $N(0, 10^{-4})$ (c) OMP result, $\text{MSE}=0.86035$ (d) ROMP result, $\text{MSE}=0.56895$ (e) StOMP result, $\text{MSE}=0.38354$ (f) SP result, $\text{MSE}=0.26936$ (g) BP result, $\text{MSE}=0.20166$ (h) BAOMP result, $\text{MSE}=0.0022514$.

In Experiment 7 and 8, we examine the average approximation performance of the proposed method for the signal perturbation and measurements perturbation cases, respectively. Like the above example, the non-zero entries of signals were generated from standard Gaussian distribution $N(0,1)$, and the noise variance was set as $\delta_s^2 = 10^{-4}$ or $\delta_z^2 = 10^{-4}$, and each (K, N, M) case repeated 500 times to obtain the average reconstruction error $\sum (\mathbf{x}_\Lambda - \mathbf{x}_\Lambda^{\text{rec}})^2 / K$. The proposed method used $\mu_1 = 0.4, \mu_2 = 0.6, n_{\max} = M, \varepsilon = 8N\delta_s^2$ or $8N\delta_z^2$ as the inputs. All other methods also used $\varepsilon = 8N\delta_s^2$ or $8N\delta_z^2$ as a parameter to control the convergence. Figure 4.10 and Figure 4.11 give the performance results for the two perturbations cases, respectively.

(a) Gaussian case: the average reconstruction error for signal perturbation case

(b) Binary case: the average reconstruction error for signal perturbation case

Figure 4.10: Simulations of the average reconstruction error for signal perturbation case vs. number of measurements M by using different methods: (a) Gaussian sparse signal (b) binary sparse signal.

As shown in Figure 4.10(a), for the signal perturbation case, our proposed method gives the lowest average reconstruction error among all the used methods for

Gaussian sparse signal recovery except the cases when $M < 70$; with the number of measurements $M > 120$, SP method also gives similar performance to our proposed method. For the binary sparse signal approximation, as shown in Figure 4.10(b), with $M < 110$, BP gives the best performance. But with $M > 110$, the proposed method gives the best reconstruction results, and when the number of measurements is $M \geq 140$, the proposed method and SP give similar results. However, SP method requires using the sparsity level K in its process and the proposed BAOMP method does not need this information. Figure 4.11 demonstrates the average reconstruction error for measurement perturbation case; it shows a similar performance as shown in Figure 4.10. Furthermore, from all the above figures for noisy cases, we find that when the number of measurements is big enough such as $M > 130$, the average reconstruction error of the proposed method is between 10^{-3} and 10^{-4} for all the cases, which is nearly the same as the order of the noise variance $O(10^{-4})$.

(a) Gaussian case: the average reconstruction error for measurement perturbation case

(b) Binary case: the average reconstruction error for measurement perturbation case
 Figure 4.11: Simulations of the average reconstruction error for measurement perturbation case vs. number of measurements M by using different methods: (a) Gaussian sparse signal (b) binary zero-one sparse signal.

All the above experiments in this subsection show the proposed BAOMP method is effective for sparse reconstruction and approximation from noisy signals or noisy measurements. Hence, it demonstrates that the proposed algorithm is robust and broadly applicable to many different sparse signal processing areas.

4.3.4 Experiments on 2-D Compressible Signal Reconstruction

In this subsection, the final experiment compared the performance of some of the above mentioned algorithms for practical image processing. Here we used StOMP, SP, ℓ_1 -norm minimization methods and the proposed method to obtain the reconstruction. GPSR method [122] was used as the ℓ_1 -norm minimization method since it has a good trade-off between fast reconstruction and good performance

among the ℓ_1 -norm optimization methods. A test image, taken from part of “Lena” image, was used in our experiment. Firstly, the test image was partitioned into $N = 64 \times 64$ blocks; for each block, we performed DCT transform and quantized it with quantization value 32 to obtain a K -sparse coefficient representation, and then used a sampling matrix Φ generated from standard Gaussian distribution to collect the measurements. At the reconstruction side, firstly the above mentioned reconstruction methods were used to reconstruct the K -sparse coefficients, which were then de-quantized and IDCT transformed to recover each block. Finally the whole image was reconstructed. Figure 4.12 illustrates the reconstructed images from $M = \min(\text{round}(3.5K), N)$ measurements for each block.

(a)Original image

(b) DCT reconstruction, PSNR=32.5091dB

(c)StOMP rec., PSNR= 29.4375dB

(d) Rec. error per pixel for StOMP method

(e) SP rec., PSNR= 29.2213dB

(f) Rec. error per pixel for SP method

(g) GPSR rec., PSNR=29.8603dB

(h) Rec. error per pixel for GPSR method

(i) BAOMP rec., PSNR= 31.4372 dB

(j) Rec. error per pixel for BAOMP method

Figure 4.12: Reconstructed image from $M = \min(\text{round}(3.5K), N)$ measurements: (a) the original image (b) reconstructed image by DCT with the best K -sparse coefficients, PSNR = 32.5091 dB (c) reconstructed image by StOMP method, PSNR = 29.4375dB (d) reconstruction error per pixel for StOMP method (e) reconstructed image by SP method, PSNR = 29.2213 dB (f) reconstruction error per pixel for SP method (g) reconstructed image by GPSR ℓ_1 -norm minimization method, PSNR = 29.8603 dB (h) reconstruction error per pixel for GPSR method (i) reconstructed image by the proposed BAOMP method, PSNR = 31.4372 dB (j) reconstruction error per pixel for BAOMP method.

As can be seen in Figure 4.12, the PSNR value of the best K -term DCT coefficient representation is 32.5091 dB. While other methods result in performance degradations of 2.65 dB to 3.28 dB, the proposed method gives the best performance among all the tested methods with performance degradation of just 1 dB. Besides, as shown in the figures of reconstruction error per pixel, the proposed method give the least reconstruction error per pixel in all the CS reconstructed figures. These figures show the subjective quality of the image reconstructed by BAOMP method to be superior to other methods (note the fine texture on the hat and the reconstruction error per pixel figure). This experiment shows the proposed method is also effective for the compressible signal's reconstruction and approximation.

In summary, from all the above experiments, the proposed BAOMP method outperforms all the other tested OMP-type methods, including OMP, ROMP, StOMP, and SP algorithms. Furthermore, for the sparse signal with Gaussian-distributed non-zero coefficients, the proposed method even gives better performance than BP method. With computational complexity comparable to other OMP-type methods, the proposed BAOMP method can obtain comparable or even better reconstruction performance to that of the above two ℓ_1 -norm optimization methods. Therefore, we believe this method will be useful and effective in the field of CS and other sparse reconstruction and approximation areas, and will certainly improve the reconstruction quality with low computational cost.

4.4 Summary

In this chapter a novel OMP method, called Backtracking-based Adaptive OMP, is

proposed for CS reconstruction and other sparse signal reconstruction and approximation problems, with the following distinguishing features:

- 1) Backtracking step involving a preliminary selection and a final selection into OMP method, using which the proposed method could flexibly remove some atoms that are chosen wrongly at the previous iterations;
- 2) No prior knowledge of sparsity level K is required; an adaptive threshold based on μ_2 effectively continues the iterations until the correct K .

Examples show how BAOMP backtracking improves upon OMP performance, and how in presence of noise BAOMP method reconstructs better than all the other considered methods.

Extensive simulation results comparing BAOMP method with BP, OMP, ROMP, StOMP, SP and GPSR methods show the following:

- 1) In the 1-D exact sparse signal reconstruction case, for Gaussian sparse signal BAOMP method has the best performance, with higher critical sparsity and significantly higher probability of exact reconstruction with inadequate measurements. For binary sparse signal (which is known to be difficult for OMP-type methods since all the non-zero values in this signal have equal energy and hence OMP-type methods could be easily trapped in some wrong local minimum points), BAOMP method outperforms all the other OMP-type methods, but it is worse than BP method. However, BP method has significantly higher complexity than BAOMP method;
- 2) The average number of iterations of BAOMP method required for perfect reconstruction is much less than the sparsity level K , is mostly insensitive to

the number of measurements M , but is dependent on the parameters μ_1 and μ_2 ;

- 3) The computational complexity of BAOMP method is enumerated and is shown to be comparable to OMP method for very sparse signals;
- 4) Best μ_1 and μ_2 values for exact reconstruction of Gaussian and binary sparse signals are experimentally determined;
- 5) In presence of signal and measurement perturbations, for Gaussian sparse signal case, BAOMP method performs the best except at very few measurements. For binary case, BAOMP method is the best with more measurements while BP method is the best with fewer measurements. In both cases SP method performs comparably at large number of measurements; however, SP method uses exact knowledge of K , without which it will perform worse. Note that with sufficient measurements, the reconstruction error of BAOMP method is comparable to the perturbation noise level;
- 6) For compressible signal (image) reconstruction, BAOMP method outperforms GPSR ℓ_1 -norm optimization, StOMP and SP methods both objectively and subjectively. It performs only about 1dB below the best K -term reconstruction case, and up to 2.5dB better than other methods.

Therefore, it may be concluded that the proposed BAOMP method is superior to all other considered methods, and is suitable for a range of sparse reconstruction and approximation problems and applications. For example, it is very appropriate for some sparse signal reconstruction applications where the sparsity level K is unknown and some greedy algorithms (like SP) may not get good performance with an estimated K .

Since the sparsity level is an unknown parameter, one drawback of the proposed method is that currently we cannot easily extend the proofs of SP and CoSaMP algorithms to our case to give provable performance guarantees. This is one problem that needs further research.

Chapter 5

Object-based Surveillance Video Compression using CS

5.1 Introduction

Video surveillance is one of the most active research areas in recent years. One main aspect of the surveillance system is to efficiently compress the huge volumes of recorded video, to facilitate subsequent processing. However, this aspect has not been sufficiently explored as compared to the other modules in the surveillance system. Efficient video compression is very important for any surveillance system since most of the surveillance videos are recorded continuously 24 hours a day and stored for a period ranging from few days to many months. In such situations, it is essential to have efficient compression that can greatly reduce the storage space requirements. Today, most surveillance systems are network-connected for centralized real-time monitoring. The efficient compression techniques also play a crucial role in such network-connected surveillance systems to efficiently utilize the available bandwidth for real-time monitoring via various network protocols.

Although there have been many video technologies and compression standards, it is hard to find a dedicated system for efficient coding of surveillance

videos. Since the main goal of most video surveillance systems is to detect the foreground objects and the backgrounds nearly have no change in the whole surveillance period, object-based coding techniques may be more suitable for efficient surveillance video compression than other kinds of coding schemes, such as the block-based coding standard H.264 [123]. The object-based video compression scheme has long been recognized within Moving Picture Experts Group (MPEG) community as having the potential to transmit high quality video at substantially lower bit rates than that afforded by its more well-established block-based DCT transform coding schemes. There have been few reported works based on MPEG4 object-based coding methods for surveillance video coding [124, 125]. However, most of the object-based coding techniques have two major drawbacks. Firstly, object segmentation and motion estimation methods are computationally intensive. Secondly, accurately representing the shapes of moving objects would result in insufficient number of bits for coding the content of a low bit rate encoder. In this chapter, we also explore object-based video compression method to address the issue of efficient surveillance video compression. In our proposed method, we use a simple segmentation method to segment the foreground moving object, then the motion estimation is just done based on the whole object, and finally the object error is coded by simply projecting on a CS random sampling matrix. The whole system is simple at the encoder side, and the experiments prove it is effective for surveillance video compression.

In this chapter, we use CS framework to encode the object error resulted after object segmentation and motion compensation. Because of high temporal correlation of video frames, the resulting object error after motion compensation is often really

sparse in the whole frame or in an object block that includes the moving object. Hence, CS approach could be adapted to the object-based video compression scheme to encode the object error. The experimental results prove its efficiency.

This chapter is organized as follows: In Section 5.2, the proposed surveillance video compression scheme based on CS is presented in detail. The experimental results are performed and discussed in Section 5.3. Finally, we give the conclusions in the last Section 5.4.

5.2 Object-based Video Compression Scheme

With emerging wireless and mobile applications, there was considerable research in the area of object-based coding for very low bit rate compression [124, 125]. The object-based compression is achieved by separating the foreground moving objects from stationary background and compactly representing their shapes, motions and contents. In this chapter, the proposed object-based video compression scheme includes four major steps: object segmentation, object-based motion estimation and compensation, CS-based object error coding, and entropy coding. The process of the proposed coding technique is illustrated in Figure 5.1. At the encoder side, the foreground moving objects are firstly segmented from the background, then the moving objects are object-based motion compensated from the previous reconstruction frame, and finally the resulting object error is encoded by some CS random matrix and is quantized before entropy coding. At the decoder side, the encoded stream is firstly de-quantized to recover the CS measurements, then the object error blocks are recovered from these CS measurements by using some CS reconstruction methods,

and finally the reconstructed frames are generated by adding the recovered object error, the background and the motion compensation together. The details are described in the following subsections sequentially.

(a) The Proposed Encoder

(b) The Proposed Decoder

Figure 5.1: The block diagram of the proposed compression scheme: (a) the proposed encoder (b) the proposed decoder.

5.2.1 Object Segmentation

In object-based compression scheme, the first step is the object segmentation. It is one of the important processes in object-based compression scheme. Better segmentation will greatly reduce the bits required to encode and transmit the video sequence. In this chapter, to perform a simple and fast segmentation, we use the edge-based object segmentation technique proposed in [126] to separate the foreground moving object from the background.

The edge-based segmentation technique first uses some image attributes, such as object pixels, object edge map and object difference edge map, to separate the objects. Then it uses morphological operations to extract the foreground objects. The background image is assumed known a priori in the compression system since most surveillance systems have fixed cameras and the background have small change in the whole surveillance period. For the case of the unknown background in the decoder side, it could be encoded as an intra frame by H.264 using high bitrates or could be well constructed using some appropriate background generation methods [127, 128]. The major steps of the object segmentation are given as follows:

- 1) Object pixels $\mathbf{O}$ are obtained by subtracting the current frame from the background image.

$$\mathbf{O}(i,j) = \begin{cases} 1, & \text{if } |\mathbf{f}_n(i,j) - \mathbf{bg}(i,j)| > \tau \\ 0, & \text{otherwise} \end{cases} \quad (5.1)$$

Where, $\mathbf{f}_n$ and $\mathbf{bg}$ represent the current and background frames respectively. τ is a threshold to distinguish the moving object from background. The selection of an appropriate threshold is important because a low threshold could result in over-

segmentation, while a high threshold could result in under-segmentation and incomplete objects. Since it is not necessary to produce a precise segmentation of the moving objects in terms of gaining high coding efficiency in our system, we just set 6% of the maximum illumination value as the threshold in our experiments. In order to make the operation robust, the foreground binary image mask is obtained as the aggregation of object pixels with all ‘on’ pixels of the object edge map $\mathbf{E}_O = |\mathbf{e}(\mathbf{f}_n) - \mathbf{e}(\mathbf{bg})|$ and the different edge map $\mathbf{E}_d = |\mathbf{e}(\mathbf{f}_n) - \mathbf{e}(\mathbf{f}_{n-1})|$,

$$\mathbf{M}_n(i, j) = \mathbf{O}(i, j) + \mathbf{E}_O(i, j) + \mathbf{E}_d(i, j) \quad (5.2)$$

where edge maps $\mathbf{e}(\mathbf{f})$ are obtained by Canny edge detector;

- 2) Perform ‘flood fill’ operation to fill the holes inside the object mask, and then perform binary opening and close operation to remove the noisy pixels in the mask;
- 3) The above open operation is effective for removing noise within the object region and the close operation is good for eliminating the background noise. However, noise regions whose areas are larger than the structuring element can’t be removed by the open or close operations. Hence, the noise regions with large area are removed by using the following steps:
 - (a) The objects that occupy number of pixels less than a threshold are removed. In our system this threshold is set at 100 pixels;
 - (b) The remaining objects, whose pixel density ρ is less than a threshold T_ρ is removed. ρ is computed as $\rho = \frac{\text{Number of 'on' pixels in the } k^{th} \text{ object}}{\text{Area of the } k^{th} \text{ object}}$. The value of T_ρ

is set at 0.4 in our method, which requires that at least 40% of the object pixels do not belong to the background.

Noted that all the above operations are processed on Y-plane of each frame. Figure 5.2 illustrates an example of the final segmented object mask $\mathbf{M}$ and the segmented objects.

Figure 5.2: The segmented results for one frame of the ‘walk’ video sequence: (a) aggregation of object pixels after step 1 (b) mask after morphological operations after step 2 (c) the final mask after step 3 (d) the segmented objects.

5.2.2 Motion Estimation and Compensation

Motion estimation and compensation plays a significant role in most video compression standards, and it is also a very computationally intensive part in most compression algorithms. While the traditional block-based motion estimation and compensation works on blocks for the whole frame, the object-based motion estimation and compensation is performed for each object individually, which could greatly reduce the processing time. Let $\tilde{\mathbf{f}}_{n-1}$ denote the previous reconstructed frame (for the first encoding frame, the background image can be used as $\tilde{\mathbf{f}}_{n-1}$ or one can just code the first frame in some intra coding modes, like H.264 I-frame or JPEG2000 coding), then the object-based motion estimation can be expressed as,

$$\mathbf{u}_j = \arg \min_{\mathbf{u} \in \mathbf{W}} \sum_j \mathbf{M}_j [\tilde{\mathbf{f}}_{n-1}(\mathbf{u} + \mathbf{x}_j) - \mathbf{f}_n(\mathbf{x}_j)]^2 \quad (5.3)$$

where j indicates the object number, $\mathbf{u}_j$ is the estimated motion vector for the j^{th} object, $\mathbf{M}_j$ is the binary object mask for the j^{th} object obtained from the above segmentation step, $\mathbf{x}_j$ are the locations of the j^{th} object in n^{th} frame, $\mathbf{f}_n$ is the current frame, and $\mathbf{W}$ is the search window around $\mathbf{x}_j$. Just the Y-plane is used to estimate the motion vector.

After motion estimation, the object-based motion compensation for each plane is performed as follows,

$$\hat{\mathbf{f}}_n = \mathbf{bg} \otimes \bar{\mathbf{M}}_j + \sum_j \mathbf{M}_j \otimes [\tilde{\mathbf{f}}_{n-1}(\mathbf{u}_j + \mathbf{x}_j)] \quad (5.4)$$

where $\hat{\mathbf{f}}_n$ denotes the reconstructed frame by object-based motion compensation, $\bar{\mathbf{M}}_j$ denotes binary complement of the object mask $\mathbf{M}_j$, $\mathbf{bg}$ is the background frame and

$\otimes$ represents the point-wise multiplication operation. After object-based motion estimation and compensation, the motion compensated object error Δ_j is given by

$$\Delta_j = \mathbf{M}_j(\mathbf{f}_n - \hat{\mathbf{f}}_n) \quad (5.5)$$

5.2.3 CS-based Object Error Coding

After motion compensation, the resulting object error is usually sparse in the block that includes the moving object. The object error block is denoted as the minimum rectangle widow that includes the individual object error. There will be n object error blocks if the current frame includes n segmented objects. Figure 5.3 (b) gives an example of the object error block, and it shows the object error is really sparse in the block (black pixels are the zero values). Since the resulting object error block is sparse, CS can be adapted to encode the object error and expected to get good compression performance in video compression. The proposed CS-based compression method separately processes each object error block individually for each plane of the frames.

In the CS-based object error coding step, each 2-D object error block is first reshaped into 1-D signal and then projected onto a Gaussian random matrix to get the measurements. Finally the measurements are quantized. The procedure for object error coding at the encoder side by CS framework is given as follows:

- 1) Reshape each object error block into 1-D signal $\mathbf{x}_{\text{ob}}$;
- 2) For a chosen number of measurements M , create a random matrix Φ_0 (each entry is drawn from i.i.d. zero mean and unit variance Gaussian distribution), and then obtain the sampling measurements $\mathbf{y}_{\text{ob}} = \Phi \mathbf{x}_{\text{ob}}$;

-
- 3) An independent uniform scalar quantizer with quantization value Q_t is used to quantize the measurements.

At the decoder side, the entropy decoded and de-quantized sampling measurements are recovered by using some CS reconstruction methods, like BP, OMP and StOMP algorithms. Let $\hat{\Delta}_j$ denote the reconstructed object error by using CS reconstruction method, and then the finally reconstructed frame is given by:

$$\tilde{\mathbf{f}}_n = \hat{\mathbf{f}}_n + \sum_j \mathbf{M}_j \otimes \hat{\Delta}_j \quad (5.6)$$

The choice of measurement number M has effect on the reconstruction quality and bitrates. From the experiments, we observe 1.5K random samples are enough to get acceptable reconstruction quality for most K non-zero pixels' object error blocks.

5.2.4 Quantization

The following parameters for each video frame are needed to transmit to the decoder side in the proposed CS framework:

- 1) Object mask and location information for each segmented object;
- 2) Motion vectors generated from the motion estimation and compensation;
- 3) Number of measurements M and the seed for generating the Gaussian random CS sampling matrix;
- 4) The quantized CS measurement coefficients

An independent uniform scalar quantizer with quantization value Q_t is used to quantize the CS measurements. The number of measurements M and the seed for

generating the Gaussian random matrix can be preset to the same values at the encoder and decoder to avoid transmitting. Each object mask is firstly encoded by chain code [129], and then along with the above parameters is coded using the arithmetic entropy coding to transmit.

5.3 Experimental Results and Discussions

In this section, to test the proposed surveillance video compression scheme, we have used the video sequences obtained from CAVIAR (Context Aware Vision using Image-based Active Recognition) project site [130]. The performance of reconstructed video frames using the proposed method for ‘walk’ sequence (which is a typical surveillance video sequence) will be shown in following subsections.

Firstly, we do some simulations on reconstruction of signal error block by using different CS reconstruction methods and choose the best one to use in the proposed video compression system. Then experiments on reconstruction of whole video sequence are done to show the performance for surveillance video compression. Finally, the performance is compared with H.264 compression scheme to show the proposed system’s efficiency.

5.3.1 Experiments on Reconstruction of Single Object Error Block

We take an object error block (given in Figure 5.3 (b)) as an example to show the reconstruction performance of CS method, and this block is obtained from one video

frame of the ‘walk’ sequence. The size of the object error block is 32×50 , and there are 547 nonzero pixels. Although this block is not very sparse, the reconstruction results are still good by using 1.5K random measurements. These samples are just nearly one half of the total pixels in the block. Of course we also find that by using more measurements, the reconstruction is better. Sometimes it can obtain an exact reconstruction. For example, BP and OMP methods can give exact reconstructions when 2K samples are used for this object error block. The more the random collected samples are used, the better result it gives. Since exact reconstruction is not necessary for most video applications, here we give and discuss several reconstruction results by using 1.5K samples.

In our experiment, we use three methods (BP, OMP and StOMP) to reconstruct the object error block. Furthermore, we also consider the quantization effect on each reconstruction method when the CS measurements are quantized. The reconstruction results are given in the following figures (Figure 5.3-5.6). The numerical performance is given in Table 5.1.

(a)

(b)

Figure 5.3: The original object error block segmented from the whole video frame: (a) the whole video frame (b) the resulting object error block.

(a)

(b)

Figure 5.4: Reconstruction by BP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16.

(a)

(b)

Figure 5.5: Reconstruction by OMP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16.

(a)

(b)

Figure 5.6: Reconstruction by StOMP using $1.5K = 861$ measurements: (a) reconstructed block when no quantization is used (b) reconstructed block when quantization is used with quantization value 16.

Table 5.1: The Performance Comparison of BP, OMP and StOMP Methods

Method	No Quantization		Quantization	
	<i>PSNR(dB)</i>	<i>CPU time (seconds)</i>	<i>PSNR(dB)</i>	<i>CPU time (seconds)</i>
BP	40.968	32.031	32.748	34.000
OMP	38.164	8.125	29.033	7.922
StOMP	36.836	0.484	32.873	0.219

Figure 5.3-5.6 compares the reconstructed object error blocks by using different methods. They show that all the three methods can approximate the original object error block closely with 1.5K non-adaptive measurements, especially when there is no quantization process used. These results are very important for CS-based object error coding method. When there are much fewer non-zero pixels in the object error block, the needed measurements could be much fewer than the size of the block, and hence it could achieve a higher compression ratio than other compression schemes.

Considering quantization is one of the main processing steps in the lossy signal compression schemes, we also use the independent uniform scalar quantizer with quantization value of 16 to quantize the random measurements and then reconstruct the blocks from these quantized measurements. These reconstruction results are also given in the Figure 5.3-5.6. Although the results are noisy when quantization is used, the main feature information of the block is preserved well. Observing the figures, although the reconstruction results with non-quantized samples are much better than the reconstruction results with quantized samples, the quantized samples need much

fewer bits to entropy code. These figures show the proposed CS-based object error coding method is effective even when quantization process exists.

In Table 5.1, PSNR value and CPU running time are used to evaluate each reconstruction method, and the results are corresponded to the above figures. When no quantization is used, we observe BP reconstruction can get the best reconstruction result with $\text{PSNR}_{\text{BP}} = 40.968\text{dB}$ by using 861 random measurements, the other results are $\text{PSNR}_{\text{OMP}} = 38.164\text{dB}$ and $\text{PSNR}_{\text{StOMP}} = 36.836\text{dB}$, respectively. Since BP is based on global optimization, it gets the best result with the highest PSNR value and reconstruction quality. Considering the quantization effect on reconstruction, when a quantization value of 16 is used to quantize the measurements, the PSNR values of the reconstruction blocks are $\text{PSNR}_{\text{BP}} = 32.748\text{ dB}$, $\text{PSNR}_{\text{OMP}} = 29.033\text{ dB}$ and $\text{PSNR}_{\text{StOMP}} = 32.873\text{ dB}$, respectively. StOMP method can get a slightly better reconstruction result than BP method when quantized measurements are used. Here PSNR value is just computed for the object error block. If we consider the whole frame, PSNR values can increase even more.

Although BP reconstruction is a more sophisticated approach based on global optimization and can offer the best performance most of the time, it requires fairly heavy computations and is very slow, especially for large-scale data system. From Table 5.1, BP requires much more time to get the result than OMP and StOMP. Because StOMP uses a fixed number of stages and chooses more atoms at each stage, it needs less time than OMP. Actually, we can also consider the object error in the whole frame, and use CS method to encode and reconstruct the whole frame, but it may need much more time to reconstruct, especially for BP reconstruction. Hence in

this section we don't consider this kind of reconstruction, we just consider the reconstruction case in the object block.

5.3.2 Experiments and Comparisons on Whole Video Sequence

In this section, we observe the performance for a video sequence compression. We still use the 'walk' video sequences obtained from CAVIAR project site [130]. The performance of reconstructed video frames using the proposed method for 'walk' sequence (which is a typical surveillance video sequence) is shown in Figure 5.7 when different quantization values Q_t are used. In our settings, we use $2K$ (2 times of non-zero entries) as the number of measurements and StOMP method as CS reconstruction method since it is fast and have a relatively better reconstruction quality, and then use different quantization values to process the compression. From Figure 5.7, it is shown that when Q_t is increased from 16 to 32, the required average bits/frame changes from 6139 to 4145; however, the corresponding reconstruction quality does not change too much, the average PSNR of Y-plane just marginally changes from 35.9782dB to 35.555dB. That means it can give a higher compression ratio with slight degradation of the video quality by using a higher quantization value.

To show the efficiency of the proposed compression technique, we also compare the results with the H.264/AVC standard. Figure 5.8 and Figure 5.9 show the reconstructed results by using the proposed method and H.264 JM reference software version jm10.2 [123]. The proposed method uses $2K$ measurements, $Q_t=16$, and StOMP reconstruction method to get the reconstructed results. We use the High 4:4:4 profile, 4:4:4 YUV Format, $Q_t=32$, the first frame encoded as I-frame, and all the

following frames encoded as P-frame as the configuring parameters to obtain the H.264 reconstructed result. In the following experiments, we always use these parameters except changing the quantization values for the final experiment.

Figure 5.8 gives the reconstructed results for the 5th frame. From Figure 5.8, it is easy to see our method can give a slightly better reconstruction quality than the H.264 standards. In the CS reconstructed frame by using the proposed object-based video compression scheme, the most important feature is clearly recovered; however, in the H.264 reconstructed frame, the reconstructed object is more blurry than that given by CS reconstruction. On the other hand, since our proposed method just codes the foreground object, the required bits of the proposed method is much smaller than that of H.264. For example, for this frame, it needs 10080 bits for encoding by H.264; however, our method just requires 5943 bits.

(a) Y-plane PSNR values for each frame

(b) Bits needed for encoding each frame

Figure 5.7: The reconstruction performance by the proposed method when quantization values Qt of 16 and 32 are used for 'walk' sequence with 30 frames to be coded (a) the Y-plane PSNR values of each reconstructed frame, (b) the bits required for encoding each frame.

(a) The original 5th frame

(b) The reconstructed 5th frame by using H.264 jm10.2, PSNR=34.438dB, bits=10080bits

(c) The reconstructed 5th frame by using the proposed method, PSNR=36.2908dB, bits=5943bits

Figure 5.8: The reconstructed frame by using the proposed method and H.264 jm10.2 reference software: (a) the original video frame (b) the reconstructed 5th frame by using H.264 software jm10.2, PSNR=34.438dB, bits=10080bits (c) the reconstructed 5th frame by using the proposed method, PSNR=36.2908dB, bits=5943bits.

Figure 5.9 compares the compression performance for the whole sequence (except the first I-frame) by using the proposed method and H.264 standard. As shown

in Figure 5.9 (d), when much fewer bits are used to the proposed method than H.264, the reconstructed whole frames' PSNR values (Figure 5.9(a)) obtained by the proposed method still have higher average PSNR value than H.264. But if we observe Figure 5.9(b) and (c), the reconstructed foreground objects have lower PSNR values by the proposed method, and the reconstructed background objects have higher PSNR values than H.264. However, if we increase more bits to encode each frame, the PSNR values of reconstructed foreground object will be better than H.264. Figure 5.10 shows the reconstruction result when quantization value $Qt=4$ and OMP method are used.

Here, we use the object mask gotten from the proposed method to break down the H.264 decoded frame to background part and foreground part, and then get the corresponding foreground and background objects' PSNR values, respectively. Although the background is usually fixed in most surveillance video, the background parts in each frame of the actually encoded video sequence may have small changes because of environment changes, such as light variation in the surveillance period. Since the change is small, we can directly use the assumed perfect background **bg** to replace the background part of the encoded frame, and don't use any bits to encode the background. However, because of the small difference between the assumed perfect background **bg** and the actual frame, the computed PSNR for each CS-code frame is not infinite, as shown in Figure 5.9 (c). If the assumed perfect background **bg** is encoded as the first frame, the PSNR for this frame will be infinite, since these two frames are the same. However, in our case, we start encoding the video sequence when there are some objects in the background, and all the encoded frames are not the same as the assumed background **bg**, hence the computed PSNR is not infinite.

(a) PSNR values of the whole frame (Y-plane)

(b) PSNR values of the foreground object (Y-plane)

(c) PSNR values of the background object (Y-plane)

(d) Bits needed for encoding each frame

Figure 5.9: The compression performance's comparison by the proposed method (Qt=16) and h.264 jm10.2 reference software: (a) PSNR values of the whole frame (b) PSNR values of the foreground object (c) PSNR values of the background object (d) the required bits for encoding each frame.

As shown in Figure 5.10, when nearly the similar bits are used, the reconstructed foreground objects' PSNR values (Figure 5.10(b)) obtained by the proposed method are much better than that by H.264. Observing Figure 5.10(b) and 5.10(c), it can be seen that the reconstructed foreground objects, which are the area that we are more interested in, have better reconstructed quality than the reconstructed background objects. However, for H.264 compression method, the reconstructed foreground objects have much worse quality than the reconstructed background objects.

From the above two experiments, Figure 5.9 and 5.10 show the proposed method provides a flexible approach to adaptively coding the foreground objects. When high quality is required for the foreground objects, due to the segmentation, we

can use more bits to encode and make the foreground objects have better quality than background.

(a) PSNR values of the whole frame (Y-plane)

(b) PSNR values of the foreground object (Y-plane)

(c) PSNR values of the background object (Y-plane)

(d) Bits needed for encoding each frame

Figure 5.10: The compression performance's comparison by the proposed method (Qt=4) and H.264 jm10.2 reference software: (a) PSNR values of the whole frame (b) PSNR values of the foreground object (c) PSNR values of the background object (d) the required bits for encoding each frame.

H.264 is effective for dealing with complex video with non-stationary background and complex motion. However, our proposed method is more suitable for surveillance video compression, where the backgrounds nearly have no change in the whole surveillance period. Besides, the encode side of some surveillance system may be not very powerful, and hence the simple encoding process of our proposed method may be more applicable. Therefore, we mainly consider the performance of the proposed method on nearly static background videos, such as most surveillance videos. Since H.264 may not be a legitimate comparison for this kind of surveillance video, we use run-length coding for the object error as a simple comparison with our method in the next two experiments.

In the following experiments, after the objects are segmented from the background and then motion compensated from the previous frame, the resulting

object error block are coded by CS method and run-length coding method using quantization value $Q_t=8$. Figure 5.11 compares the performances for the ‘walk’ sequence in no transmission lost and 20% transmitted coefficients lost cases. In the run-length coding method, the quantized object error is first coded by run-length coding to get the pairs (run-length, non-zero value), and then a fixed length code is used to encode the run-length pairs. For simplify, perfect synchronization is assumed for both CS and run-length method, and only 20% coefficients in random positions are assumed lost. The run-length coding uses fixed length code to avoid any error propagation due to variable length coding, such as Huffman coding. However, error propagation across frames is still possible due to motion compensation.

As shown in Figure 5.11(b), when there are no coefficients lost in the transmission, the run-length coding can give much better reconstructed quality than CS method, but as shown in Figure 5.11 (c), when there are randomly 20% coefficients lost in the transmission, the CS method gives a more steady reconstruction than run-length coding and it is more robust to transmission lost. This is simply because the run-length coding treats each non-zero value individually, if some non-zero values are lost, it is not possible to recover it from other information. However, CS method collects the measurements using some sensing matrix, and each measurement includes partial information of each non-zero value. If some measurements are lost, it is still possible to recover non-zero values from other measurements. Besides, if some of the non-zero values are lost in the current frame, the error is easier to propagate to the next reconstructed frames in the run-length coding than in CS method, and make the reconstruction degrade even more.

(a) PSNR values of the whole frame (Y-plane)

(b) PSNR values of the foreground object (Y-plane) (no transmission lost)

(c) PSNR values of the foreground object (Y-plane) (20% transmission lost)

(d) Bits needed for encoding each frame

Figure 5.11: The compression performance's comparison for 'walk' sequence by the proposed method and run-length coding ($Q_t=8$): (a) PSNR values of the whole frame (b) PSNR values of the foreground object (no transmission lost) (c) PSNR values of the foreground object (20% transmission lost) (d) the required bits for encoding each frame.

As the final experiment, we use another surveillance video sequence 'hall monitor' to do the experiment again, and the result is shown in Figure 5.12. As shown again, the run-length coding can give much better reconstructed quality than CS method when no transmission lost is involved, but CS method is more robust to transmission lost. This time, when some important non-zero values are lost, the reconstruction is degraded much fast.

(a) PSNR values of the whole frame (Y-plane)

(b) PSNR values of the foreground object (Y-plane) (no transmission lost)

(c) PSNR values of the foreground object (Y-plane) (20% transmission lost)

(d) Bits needed for encoding each frame

Figure 5.12: The compression performance's comparison for 'hall monitor' sequence by the proposed method and run-length coding ($Q_t=8$): (a) PSNR values of the whole frame (b) PSNR values of the foreground object (no transmission lost) (c) PSNR values of the foreground object (20% transmission lost) (d) the required bits for encoding each frame.

All the above experiments show our proposed method provides a possible alternative for surveillance video compression. Although the run-length coding method could give much better performance than CS method, CS method is more robust to transmission lost. By using nearly similar bitrates, CS method can give a better foreground object reconstruction than H.264. Hence, we should state that the proposed whole object-based compression method, not just the CS coding method for object error, provides a flexible alternative for surveillance video compression.

5.4 Summary

In this chapter we presented an object-based surveillance video compression method by incorporating CS framework. The performance of the proposed compression

technique is compared with that of the H.264 standard and run-length coding. The experimental results show the possibility to use CS idea in the compression framework to obtain better performance, and it is worth further investigation for surveillance video compression.

In our proposed CS-based video object error coding scheme, the encoder is much simple. The motion estimation is just done based on the whole object, and after motion compensation, we just need to project the object error onto a Gaussian random matrix and then quantize the measurements to transmit; the decoder is a little complex, by using different reconstruction methods, the complexity of the decoder is different. For example, if we use BP method to decode the signal, the proposed method is time-consuming and will be not suitable for real-time video applications, such as video surveillance. However, if we use StOMP method at the decoder, the proposed method is much faster and could be possible for real-time video compression applications. Furthermore, at the decode side, most surveillance systems may be very powerful, and they maybe allow distributed processing of the CS reconstruction, which also makes the real-time video surveillance applications possible.

The object-based compression could have other desirable properties for surveillance video, which provides the possibility to be used as a flexible alternative for surveillance video compression. Firstly, since the proposed method mainly codes the foreground object which is the most interesting part in the surveillance system, it can facilitate following surveillance applications such as detection. Furthermore, it may give a scalable coding for the foreground object by choosing different number of measurements and quantization values to offer different object quality. Otherwise,

there are some intriguing problems that could be considered further. For instance, in the proposed method, we just use the simple independent uniform scalar quantizer to quantize CS measurements. It should be interesting to consider another more effective quantizer for CS measurement quantization.

Chapter 6

Iterative Reconstruction of Sparse Signal Representation

6.1 Introduction

POCS is an iterative method to obtain the information from one domain using the information available in other domains, and it has been widely used in image restoration problems [131-133] and block-based DCT compression standards as a post-processing scheme to reduce the blocking artifacts in the compressed images [134]. In this chapter, we study an iterative signal reconstruction method for signal reconstruction and sparse signal compression. This method is one kind of POCS method. From known partial time-domain samples and partial frequency-domain coefficients, it iteratively projects the signal onto the known time-domain samples' subspaces and the known transform-domain coefficients' subspaces. Finally the signal can be recovered closely by using this iterative reconstruction scheme. Some experiments show this method can correctly and effectively reconstruct the underlying signals.

Although this iteration method is effective for signal reconstruction, it may not be effective in traditional signal compression schemes since we not only need to code

the locations and values of the transform-domain coefficients, we also need to code the known time-domain samples' locations and values. However, we consider a special application scheme of this iterative reconstruction method in the signal compression field. As an application of the iterative reconstruction method, a novel sparse signal compression scheme based on iterative reconstruction is presented in this chapter.

For a sparse signal, there are many zero values, and we can easily encode the zero values as the known time-domain samples since we only need to know the locations of zero values. Combined with the transmitted partial transform-domain coefficients, we can use the above iterative method to reconstruct the signal in the decoding part. Since there are many zero values in the signal, we can drop many coefficients and just need to code a few number of partial coefficients. Our experiments prove the randomly chosen transform coefficients are effective in reconstruction. However, to avoid encoding the location information of the transform coefficients, we can choose the first few coefficients to transmit and discard the rest coefficients, and thus reduce the bits needed to transmit. This scheme is expected to get a better compression result than most traditional methods in sparse signal compression.

The rest of this chapter is organized as follows: In Section 6.2, the projection onto convex sets method is briefly described. The proposed iterative reconstruction scheme is explained and discussed in Section 6.3. Section 6.4 simulates the experimental results on the reconstruction performance and on the sparse signal reconstruction. Finally, we give the conclusions in Section 6.5.

6.2 Projection Onto Convex Sets Method

In POCS algorithm, the relevant information is defined in terms of different convex sets. The solution of this problem amounts to finding a point in the intersection of closed and convex sets in a Hilbert space $\mathcal{H}$. The POCS algorithm, in which an initial estimate is sequentially projected onto the individual sets according to a periodic schedule, has been the most prevalent tool to solve such problems.

POCS-based methods have a major advantage in that they are flexible in utilizing known constraints of an unknown signal in reconstruction. A constraint can be applied as long as it can be described by a closed convex set. Given n closed convex sets $\mathbb{C}_i, i = 1, 2, \dots, n$, the iteration

$$\mathbf{f}_k = P_n P_{n-1} \cdots P_2 P_1 \mathbf{f}_{k-1}, \quad k = 1, 2, \dots \quad (6.1)$$

will converge to a limiting point $\mathbf{f}^*$ in the intersection $\mathbb{C}^* = \bigcap_{i=1}^n \mathbb{C}_i$ as $k \rightarrow \infty$, for an arbitrary initial element $\mathbf{f}_0$ in Hilbert space $\mathcal{H}$. In equation (6.1), P_i is the projection operator onto $\mathbb{C}_i$. If every known property of the original image $\mathbf{f}$ can be formulated into a corresponding closed convex set in $\mathcal{H}$, then such n properties will generate n well-defined closed convex sets $\mathbb{C}_i, i = 1, 2, \dots, n$. The original image $\mathbf{f}$ should be included in all the $\mathbb{C}_i$'s and also in the intersection of all the $\mathbb{C}_i$'s, so in $\mathbb{C}^*$, that is,

$$\mathbf{f} \in \mathbb{C}^* = \bigcap_{i=1}^n \mathbb{C}_i \quad (6.2)$$

Hence the problem of restoring $\mathbf{f}$ from its n properties is equivalent to finding at least one point belonging to $\mathbb{C}^*$ [135].

6.3 The Proposed Iterative Reconstruction Scheme

In this chapter, an iterative signal reconstruction method for signal reconstruction and sparse signal compression is presented. This method can be seen as one kind of POCS method. From known partial time-domain samples and partial frequency-domain coefficients, it iteratively projects the signal onto the known samples' subspace and the known coefficients' subspace. Then the signal could be recovered by using this iterative reconstruction scheme. The detailed contents are given in this section.

Consider a signal reconstruction problem from partial transform-domain coefficients and partial time-domain samples. Firstly, suppose there is a length- N vector $\mathbf{x}$. We perform some transform (like DCT) on it and get a transformed coefficients vector $\mathbf{y}$. Now assume we send a sequence $\hat{\mathbf{y}}$, consisting of only $N - P$ ($0 < P < N$) randomly chosen coefficients of $\mathbf{y}$. So there is some missing information in the received signal $\hat{\mathbf{y}}$, and the lost coefficients are set to 0 or estimated in $\hat{\mathbf{y}}$. In this case, we aim to recover the signal $\mathbf{x}$ from the received information. Obviously, just using the received information, we can't reconstruct the original signal exactly. However, we can design an iterative scheme to recover $\mathbf{x}$ approximately under the condition that if additional information about the original signal is known, for example, that we know P randomly chosen samples of the original signal.

There are several scenarios when knowledge of a few samples is readily available at the decoder. For example, the signal $\mathbf{x}$ may be a sparse signal (such as the quantized transform coefficients of some correlated source), and the zero/non-zero locations may be transmitted to the decoder. For another example, only a small part of

the signal $\mathbf{x}$ may have changed compared to its previous version (such as the next frame of video sequence), and the changed location information may be already known (from object boundary, or motion vector, in case of video compression). In this preliminary work we study the feasibility of such reconstruction method rather than considering any specific application.

We want to recover the signal of length- N with P time-domain samples and $N - P$ frequency-domain coefficients. The reconstruction scheme is designed like these steps (also seen in Figure 6.1):

- 1) Step1: Perform the corresponding inverse transform on the received sequence $\hat{\mathbf{y}}$ and get a vector $\hat{\mathbf{x}}$;
- 2) Step 2: Replace the P known samples into the corresponding positions of $\hat{\mathbf{x}}$ and get a new vector $\mathbf{x}'$. Check the convergence condition. If it is satisfied, stop the loop, output is $\mathbf{x} = \mathbf{x}'$;
- 3) Step 3: If the convergence condition is not satisfied, perform the transform on the vector $\mathbf{x}'$ and get a vector $\mathbf{y}'$. Then use the corresponding coefficients of $\mathbf{y}'$ to replace the lost coefficients of $\hat{\mathbf{y}}$, and repeat.

Iteratively perform this loop (step 1-3) until convergence.

Figure 6.1: The iterative reconstruction processing scheme.

The convergence condition may be checked by computing the MSE of some decision rules or when a maximum number of iterations is reached. In this reconstruction scheme, we have tried three kinds of decision rules to decide when the loop is terminated. The first rule, called unknown sample error (USE) rule, is comparing the current replaced vector $\mathbf{x}'$ with the previous $\mathbf{x}'$; the second rule, called iteration error (IE) rule, is comparing the current vector $\hat{\mathbf{x}}$ and the previous $\mathbf{x}'$; and the third rule, called known sample error (KSE) rule, is comparing the known samples and the corresponding positions' values of vector $\hat{\mathbf{x}}$. If MSE is smaller than some predefined thresholding value T , it means the convergence condition is satisfied and the loop is terminated.

This method can be classified as a POCS method. If we define the known samples' constraints as a closed convex set and the received coefficients' constraints as another closed convex set, this method actually projects between these two sets iteratively until convergence. From the known partial time-domain samples and partial frequency-domain coefficients, it can closely and effectively recover the underlying signal eventually.

From the experiments, we observed the first transform coefficient had a strong effect on the result. This is because the typical signals are low-pass, thus most of their energy is concentrated in the first coefficient. So when the lost coefficients include the first one, we estimate the first coefficient by using the following equation if we denote the known samples as $\mathbf{x}_{\Lambda(i)}$, $i = 1, \dots, P$,

$$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^P \mathbf{x}_{\Lambda(i)}}{P} \times \sqrt{N} \quad (6.3)$$

The use of the above equation is intuitive. Since the first coefficient can be seen as the average of all samples multiplied by $\sqrt{N}$ (that is because $\mathbf{y}_1$ can be expressed as $\mathbf{y}_1 = (\frac{1}{\sqrt{N}}) \times \sum_{i=1}^N \mathbf{x}_i = (\sum_{i=1}^N \mathbf{x}_i / N) \times \sqrt{N}$ and the sample average can be closely approximated by $\frac{\sum_{i=1}^P \mathbf{x}_{\Lambda(i)}}{P}$ for most high-correlated signals, this estimate makes sense and is a good initial approximation. Experiments show this estimation method is better than the method of replacing the first transform coefficient with zero when it is lost. However, all the remaining lost transform coefficients are simply replaced by zero at the start of iteration.

6.4 Experimental Results and Discussions

In this section, we do some experiments to show the performance of the proposed iterative reconstruction method. In the following subsection, we firstly consider the 1-D AR source's signal reconstruction case, and then use a natural image as the input signal to do some experiments. Finally, as an application of the iterative reconstruction method, a novel sparse signal compression scheme based on this iterative method is presented.

6.4.1 Autoregressive Source Reconstruction Results

To see the proposed reconstruction method's performance, we use the AR (1) source with $\rho = 0.95$ as the input signals (the average signal variance of the experimental signals is $\sigma^2 \approx 1.4$) in the following experiments. For the given signal, we perform 1-D DCT and IDCT transforms in the loop, and set the thresholding value $T =$

10^{-8} and the maximum iterations are 1000. For each pair (P, N) , we perform 1000 independent trials.

We use the above three different decision rules to do the experiments. The convergence rates (the convergence rate is defined as the percentage of the cases that the iterative methods are convergent to the small distortion $T = 10^{-8}$ before reaching the maximum iterations in 1000 trials) for different P and decision rules are given in Figure 6.2. From this figure, we find the convergence rates are above 87% for all the decision rules and different values of P . In the whole, USE rule gives the best convergence rates, and KSE rule gives the worst ones.

Since this iteration method can be convergent for most cases, now we observe the average iteration number needed for convergence. The results are given in Figure 6.3. From the experiments, we find the average iteration number is affected by the decision rule. USE rule needs the least time to reach convergence and KSE rule needs the most time to reach convergence. The average iteration numbers are between 80 and 180 for most cases and the minimum number of iterations needed is slightly larger than 80.

From the experiments, we also find the reconstruction results are affected by the thresholding value T . If T is bigger than 10^{-8} , the convergence rates are higher and the average iteration numbers are smaller. For example, when $T = 10^{-4}$, the convergence rates are bigger than 98%, and the average iteration numbers are smaller than 70 for all decision rules. Especially the average iteration numbers are smaller than 12 for USE rule, and the convergence rates are nearly 100% for each value of P .

Although the results are convergent in some case, the reconstruction error may be much larger than the predefined thresholding value ($T = 10^{-8}$ in our experiments) since we can't directly compare MSE between the original signal and the reconstructed one. The final experiment in this subsection is to check the average reconstruction error in MSE between the original signal and the reconstructed one for the different decision rules. The experimental results are given in Figure 6.4.

Figure 6.2: The convergence rate (%) for three decision rules when $N=8$.

Figure 6.3: The average iteration number needed for convergence by the three decision rules when $N=8$.

Figure 6.4: The average reconstruction error of convergence cases for three decision rules when $N=8$.

From Figure 6.4, we find the reconstructed signal is not exactly the same as the original one for some convergence cases. For most cases the reconstruction error is between 0.01 and 0.025. That means for some convergence cases, although the reconstruction can't converge to the original one, it still can give an approximate reconstruction.

6.4.2 Image Reconstruction Results

In this subsection, we use a natural image as the input signal to do some experiments to show the proposed method's performance. For image reconstruction case, we firstly partition the image into small blocks of size 8×8 , and then reshape each block into a 1-D vector. Iterative reconstruction of each block is then obtained from

partial image samples and partial coefficients. Finally, we merge the blocks together and get the final reconstructed image.

Since image data involve integers, the iterations may be carried out using integers (rounding off the vector $\hat{\mathbf{x}}$ in the loop) or using real numbers. Our studies show that integer iterations hamper the convergence performance. Therefore, we don't use integers during the iteration loops. Only after convergence or maximum iteration number is reached, $\hat{\mathbf{x}}$ is rounded to the nearest integers. In our experiments for image reconstruction, the maximum iteration number is 100, and the thresholding value is $T = 0.1$ when image pixels have range 0 to 255.

From the experiments, we observe that even though there are some cases when the reconstructed blocks are not exactly the original ones, the reconstructed image is still very close to the original one. For example, taking the image 'cameraman' as input, the reconstructed images by using the proposed iterative scheme are given in Figure 6.5 ($P = 10$ for each block). To show our proposed method's efficiency, we also compare the reconstructed results with those by using other non-iterative methods (in absence of any known iterative or POCS-based method for this scenario). The first method is that we just use the known transform coefficients and pad the lost coefficients with zero to do inverse transform to get the result (given in Figure 6.5 (b)). The second method is that we pad the lost coefficients with zeros to do inverse transform and then replace the corresponding samples with known samples to get the result (given in Figure 6.5 (c)). The results of these two methods are much worse than the results of the proposed iterative reconstruction method by using the three decision rules (Figure 6.5 (d)-(f)), especially from visual observation.

Figure 6.5: The reconstructed images by different methods when $P = 10$: (a) the original image, (b) reconstruction by padding the lost coefficients with zeros, PSNR=31.002dB, (c) reconstruction by padding the lost coefficients with zeros and then replacing the corresponding samples with known samples, PSNR=31.741dB, (d) reconstruction by USE rule, PSNR=38.672dB, (e) reconstruction by IE rule, PSNR=41.396dB, (f) reconstruction by KSE rule, PSNR=41.385dB.

On the whole, PSNR values of the reconstructed images for different P are given in Figure 6.6. From this figure, it is observed that the iterative method can effectively reconstruct an image with high PSNR value no matter how many coefficients are lost in each block, just requiring the corresponding number of time-domain samples to be known.

Furthermore, we also consider PSNR changes with iterations when $P = 10$ for the image ‘cameraman’, and the results are given in Figure 6.7. From the experiment, we find PSNR increasing with iterations, and merely 10 iterations suffice to achieve a high PSNR value for the image. When iteration number is smaller, PSNR value increases much more than when iteration number is bigger. Thus, it shows that the proposed reconstruction method can iteratively improve the reconstructed image's quality.

Figure 6.6: The PSNR values of reconstructed images for different P .

Figure 6.7: Convergence plots of the iteration method for image reconstruction.

From all the above experiments, the IE reconstruction result by IE rule has a high convergence rate, a low average iteration number and a high PSNR value, and it gives the best reconstruction performance considering the convergence rate, the average iteration number and PSNR values of the reconstructed images.

6.4.3 Results for Sparse Signal Reconstruction

In the above subsections, we showed the proposed method is effective for signal reconstruction. Although it is effective for signal reconstruction, it may not be effective in traditional signal compression schemes since we not only need to code the locations and values of the coefficients, we also need to code the known samples' locations and values. However, it may be useful for sparse signal compression. As a special application of the iterative reconstruction method, a novel sparse signal reconstruction scheme based on the iterative method is presented in this subsection.

For a sparse signal, there are many zero values, and we can easily encode the zero values as the known time-domain samples since we only need to know the locations of zero values. Combined with the transmitted partial transform-domain coefficients, we can use the above iterative method to reconstruct the signal in the decoding part. Since there are many zero values in the signal, we can drop many coefficients and just need to code a few number of transform coefficients. Our above experiments prove the randomly chosen transform coefficients are effective in reconstruction. However, to avoid encoding the location information of the transformed coefficients, we can choose the first K coefficients to transmit and discard the rest coefficients (for a K -sparse signal), and thus reduce the bits needed to transmit.

As an example, we still consider encoding the CAVIAR video object error blocks given in last chapter, and propose an object error coding scheme based on the iterative reconstruction method. In our setting, for a K -sparse signal, assume we know the zero values' positions as the known time-domain samples, and then choose the first K DCT transformed coefficients as known information. Finally we use the above iterative method to reconstruct the signal at the decoding part. From the experiments, we observe just K transform-domain coefficients are enough to get an acceptable signal recovery quality in a few iterations for most N -length K -sparse signals.

We take a video object error block as an example. The reconstruction results are given in Figure 6.8, 6.10, and 6.11. The original testing error block size is 32×50 , and there are 547 non-zero object pixels. From the experiments, we observe the

proposed iterative method can get a reconstructed image with PSNR=34.958dB by 10 iterations when using 547 non-quantized DCT coefficients. The reconstruction results are 35.427dB and 36.040dB by 100 and 1000 iterations, respectively. When no quantization is used to quantize the transmitted coefficients, the reconstruction result can be improved iteratively by using more iterations (see Figure 6.11 (a)). Furthermore, this kind of reconstruction is also very quick, for example, 1000 iterations just need about 3 seconds. From the experiments, we also find that we could get a better result when the transmitted coefficients are randomly chosen.

In the experiment, we also consider the quantization effects on reconstruction. When quantization is used, given the quantized value of a transformed coefficient $T(\mathbf{y}) = \hat{\mathbf{y}}_q$, the constraint should be $|\mathbf{y}' - \hat{\mathbf{y}}_q| < q/2$ (here q is the quantization stepsize). Hence, when the transformed coefficients are quantized, we changed the above iterative method to use this constraint, as shown in Figure 6.9. In the step 3, we modified the original reconstruction scheme into that if $|\mathbf{y}' - \hat{\mathbf{y}}_q| < q/2$, then we don't replace the lost coefficient by $\mathbf{y}'$, otherwise we use $\mathbf{y}'$ to replace the P lost coefficients of $\hat{\mathbf{y}}_q$. Finally, we did the experiment again, and the results are given in Figure 6.10 and Figure 6.11.

Figure 6.8: Reconstruction results: (a) the original object error block (b) reconstruction by 10 iterations, PSNR=34.958dB (c) reconstruction by 100 iterations, PSNR=35.427dB (d) reconstruction by 1000 iterations, PSNR=36.040dB.

Figure 6.9: The modified iterative reconstruction processing scheme.

Figure 6.10: Reconstruction results with quantization: (a) reconstruction with quantization value 8 by 10 iterations, PSNR=34.189dB (b) reconstruction with quantization value 16 by 10 iterations, PSNR=33.505dB.

Figure 6.11: The PSNR values changing with iterations: (a) no quantization is used (b) quantization value 8 is used (c) quantization value 16 is used.

As shown in these figures, with quantization, convergence is achieved quickly, but the reconstruction quality is degraded. As shown in Figure 10, when quantization value 8 is used, the reconstruction results are 34.189dB; and when the quantization

value is 16, the reconstruction PSNR is 33.505dB. As shown in Figure 11, when quantization is used, more iterative times do not increase the reconstruction quality. From all the above experiments, we find even though the transformed coefficients are quantized, we can still get a good reconstruction quality by using very few iterations.

If we compare this iterative reconstruction method with CS-based object error coding scheme presented in last chapter, the result may be better than that of CS-based method. The best result of CS-based method is StOMP method's result, which is PSNR=32.873dB when quantization value 16 is used. This method needs 861 quantized measurements to transmit. However, for the iterative reconstruction method, the reconstruction result is 33.505dB when quantization value 16 is used by 10 iterations. This iterative method needs to transmit 547 quantized coefficients and the locations of zero pixels. If we could find a good method for encoding the zero values' locations, this iterative reconstruction method expects to achieve a better compression result with fewer bits than CS-based coding scheme for sparse signal compression. One simple and possible scheme for encoding the zero values' locations is that it could denote the zero value with 1-bit symbol '0' and denote the non-zero value with symbol '1', and then use some entropy coding scheme (such as run-length coding) to encode the bit sequence. Considering the above object error block, we just need 1600 bits (200 bytes) to present the zero values' location information, and then using some entropy coding scheme can reduce the bits even more. Of course, the effective location coding schemes require more detailed research in the future.

6.5 Summary

In this chapter, we have proposed a new iterative reconstruction method for signal reconstruction and sparse signal compression. This method can be classified as one kind of POCS method. From known partial time-domain samples and partial transform coefficients, it iteratively projects the signal onto the known samples' subspaces and the known coefficients' subspaces. We have performed some experiments to show this method can correctly and effectively reconstruct the underlying signals. We use three decision rules to check the convergence and stop the iteration processing. From the experiments, the method by using IE decision rule can give the best reconstruction performance, considering the convergence rate, the average number of iterations needed and PSNR values of the reconstructed images. And it can be recommended to use in practice. As an application of the proposed iterative reconstruction method, a novel sparse signal compression scheme based on this iterative reconstruction is presented in this chapter. This scheme is expected to get a better compression result than most traditional methods in sparse signal compression.

Another possible application of the proposed iterative method in object-based video compression could be considered in the following situation, where only a small part of the signal $\mathbf{x}$ may have changed compared to its previous version (such as the next frame of video sequence), and the unchanged background may be already known (from object boundary or other ways). And hence in this case we just need to send

very small number of coefficients, and it is expected to correctly reconstruct the original signal.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

As a special area of digital signal processing field, sparse signal processing and representation has wide applications such as denoising, restoration, deblurring, inpainting, deconvolution, super-resolution, compression, blind source separation, compressive sensing, and more. Compressive sensing is a newly emerged branch generated from sparse signal processing and representation. This thesis has mainly investigated some effective sparse signal processing schemes and approaches to deal with a sparse signal or a compressible signal that is sparse on some dictionary. In addition, we also consider adapting CS approach and another iterative method into object-based video compression field to deal with the sparse object error block coding. The specific contributions of this dissertation are summarized below.

7.1.1 Optimized Matrix for Compressive Sensing

In this thesis, we have investigated the optimization of measurement matrix for better CS reconstruction performance. We began by introducing some concepts that a good designed measurement matrix should be satisfied, and then based on the optimized method proposed by Elad, we modified this scheme by using a simple polynomial

function to down scale the inner products between each columns of sampling measurement matrix, and also using a detector to estimate the convergence to stop the iterations early. The superiority of the proposed approach over Elad's method is demonstrated through some simulations on exact sparse signal reconstruction case and distributed CS case. The experimental result reveals this optimized matrix leads to better CS reconstruction performance with low extra computational cost.

7.1.2 Backtracking-based Adaptive OMP Algorithm for CS

To solve CS and other sparse signal reconstruction and approximation problems, the backtracking-based adaptive orthogonal matching pursuit method was proposed in this thesis. The proposed BAOMP method incorporates a backtracking step to more accurately choose the reliable atoms, and at the same time, the sparsity level of the underlying signal is not required to be known before reconstruction. Because of the backtracking technique, the proposed method could check the previous chosen atoms' reliability and then delete the unreliable atoms in the iterations. Furthermore, because BAOMP method doesn't need the sparsity level to be known a priori, it may be suitable to be used in some applications, in which this kind of information may be not available before reconstruction.

We do various experiments on different application cases, such as exact sparse signal reconstruction, noisy signal or noisy measurement approximation, and 2-D compressible signal approximation (like natural images). The extensive simulation results demonstrate that the proposed BAOMP method could give a much better reconstruction performance than other tested OMP-type methods such as OMP,

StOMP, ROMP, and SP, and in some cases it is even better than BP ℓ_1 -norm minimization algorithm.

7.1.3 Video Object Error Coding Method based on CS

We explored the applications of CS technique by integrating it into an object-based surveillance video compression system. In object-based video compression system, because of the high temporal correlation of video frames, after motion estimation and compensation, the resulting object error is often really sparse in the block that includes the corresponding moving object. Hence, CS approach could be adapted to this compression system to encode the object error.

In order to effectively compress the object error blocks in the video sequences, a video object error coding method based on CS is proposed in this thesis. The proposed system first assumes the moving objects have been segmented from background image and object-based motion compensated from the previous reconstructed frames, then uses Gaussian random matrix to collect the measurements of the sparse object error, and finally the collected CS measurements could be quantized to store or transmit. At the decoder, the inverse process is done. Experimental results on surveillance video compression show that the proposed video object error coding approach based on CS could outperform the popular H.264 compression standard for surveillance video coding and is also more robust to transmission lost than run-length coding. Hence, we should state that the proposed whole object-based compression method, not just the CS coding method for object error, provides a flexible alternative for surveillance video compression.

7.1.4 Sparse Signal Reconstruction based on Iterative Method

To effectively code the video object error generated in segmentation-based video compression scheme, a simple iterative method based on projection onto convex sets is designed. This method is one kind of POCS method. From known partial time-domain samples and partial frequency-domain coefficients, the proposed method iteratively projects the signal onto the known time-domain samples' subspaces and the known transform-domain coefficients' subspaces. Finally the signal can be recovered closely by using this iterative reconstruction scheme.

Firstly, we show this method can correctly and effectively reconstruct the underlying signals through extensive simulation. Then as a special application of the iterative method, a novel sparse signal reconstruction scheme based on the iterative method is presented in this thesis.

In the sparse signal application, since there are many zero values, we can easily encode the zero values as the known time-domain samples since we only need to know the locations of zero values. Then combined with the transmitted partial transform-domain coefficients, we can use the iterative method to reconstruct the signal in the decoding part. Since there are many zero values in the signal, we could drop many coefficients and just need to code a few number of transformed coefficients. Furthermore, to avoid encoding the location information of the transformed coefficients, we could choose the first few coefficients to transmit and discard the rest coefficients, and thus reduce the bits needed to transmit. This method provides some

possibility to consider how to use the sparse representation property to effectively compress the video signal.

7.2 Future Works

Based on the research in this thesis, we think the following directions may be interesting for further investigation.

- **Theoretical link between average mutual coherence and CS performance**

Studies in this thesis have shown optimizing CS sampling measurement matrix by reducing the average mutual coherence iteratively could lead to a much better CS reconstruction performance. One possible direction for future research is to develop a theoretical link between the average mutual coherence and CS reconstruction performance, and hence could theoretically prove the proposed optimized method is effective for CS reconstruction. Most existing theoretical works mainly consider the CS reconstruction behavior from worst-case stand-point such as mutual coherence and RIP property. However, in a particular application situation, the worst-case proof could not justify the actual behavior of CS reconstruction for a specific CS measurement matrix. Although Elad [80] and we provided some effective approaches for optimizing the measurement matrix for CS reconstruction, it is still expected that one could develop a theoretical link between average mutual coherence and CS reconstruction performance to justify the proposed optimization processing.

- **Provable guarantee of BAOMP method for CS sparse reconstruction**

In this thesis, a novel BAOMP method is proposed for CS and other sparse reconstruction problems. We provide extensive experimental simulations for various

sparse signal reconstruction cases to prove the proposed method's efficiency for CS sparse signal reconstruction, and also intuitively explain why the proposed method works for sparse signal reconstruction and how it works. However, since the sparsity level of the underlying signal is an unknown parameter in our case, it is not easy to extend the proofs of SP and CoSaMP algorithms to the proposed method to give a provable performance guarantee for sparse signal reconstruction. Hence, a possible extension of the current work is to further investigate the proof of the proposed method for CS reconstruction.

- **An efficient object-based compression system based on CS**

In this thesis, we proposed an application of CS approach on object-based video compression system. Although [136] talked about some negative results for CS approach used in compression field, our application approach has some differences from the assumption of that theoretical work. [136] said that the compression technique of spending $\log_2 \binom{N}{K}$ bits to communicate the sparse pattern and the remaining available bits $KR - \log_2 \binom{N}{K}$ to scalar quantize the non-zero entries of sparse signal $\mathbf{x}$ is better than the compression techniques of simply scalar quantize each component of $\mathbf{x}$ with $\frac{KR}{N}$ bits or compressive sampling compression with OMP and lasso reconstruction. The following points list some differences: firstly, [136]'s result is available for middle to high rate compression since it needs at least $\log_2 \binom{N}{K}$ bits to communicate the sparse pattern, but in our case we consider very low bit rates to compress the object error; secondly, [136] seems more caring about exact recovery of the sparse pattern, in our case the recovery of most significant

entries are of interest. Besides, our proposed system uses StOMP method to reconstruct the object error, which is more effective for noisy case or approximate reconstruction. Although the experimental results show the proposed method is more effective than H.264 for surveillance video compression, the theoretical analysis is still an open question needed to investigate further. Furthermore, how to improve the CS coding performance is still an opening question since the run-length coding can give much better performance than CS method when no transmission lost is involved.

- **A complete object-based compression system based on iterative reconstruction**

We also consider an iterative reconstruction method for object error compression in this thesis. This method is similar to POCS method. We have performed some experiments to show it could be used to encode the sparse object error. It is expected to extend the work to formulate a more sophisticated complete object-based compression system based on the iterative algorithm. To do this, the first problem is how to effectively encode the zero positions of sparse signals. Hence, it is interesting to study in more detail some sparse location encoding methods such as run-length coding and quad-tree based coding in the future work.

Author's Publications

Journal Papers

1. Honglin Huang and Anamitra Makur, "Backtracking-based Matching Pursuit method for Sparse Signal Reconstruction," *IEEE Signal Processing Letters*, vol. 18(7), pp. 391-394, Jul. 2011.
2. Honglin Huang, Anamitra Makur and Divya Venkatraman, "Object-based Surveillance Video Compression using Compressive Sensing," submitted to *IEEE JETCAS Special Issue for Compressed Sensing*.
3. Honglin Huang and Anamitra Makur, "Backtracking-based Adaptive Orthogonal Matching Pursuit Algorithm for Compressive Sensing," to be submitted to *Signal Processing*.
4. Honglin Huang and Anamitra Makur, "Optimized Measurement Matrix for Compressive Sensing," to be submitted.

Conference Papers

1. Honglin Huang and Anamitra Makur, "Optimized Measurement Matrix for Compressive Sensing," *2011 9th Int. Conf. on Sampling Theory and Applications (SampTA)*, May 2011. Available online: <http://sampta2011.ntu.edu.sg/proceedings.asp>
2. Honglin Huang and Anamitra Makur, "An adaptive OMP-based compressive sensing reconstruction method for unknown sparsity noisy signals," *2011 9th*

Int. Conf. on Sampling Theory and Applications (SampTA), May 2011.

Available online: <http://sampta2011.ntu.edu.sg/proceedings.asp>

3. Honglin Huang and Anamitra Makur, "A New Iterative Reconstruction Scheme for Signal Reconstruction," *2008 IEEE Asia Pacific Conf. on Circuits and Systems (APCCAS 2008)*, pp. 336-339, Macao, China, Nov. 2008
4. Honglin Huang, Anamitra Makur and Divya Venkatraman, "Video object Error Coding Method based on Compressive Sensing," *2008 10th Int. Conf. on Control, Automation, Robotics and Vision (ICARCV 2008)*, pp.1287-1291, Hanoi, Vietnam, Dec. 2008.

Author's publications not related to this PhD work

1. Jin Wu, Honglin Huang, Jinwen Tian and Jian Liu, "Remote Sensing Image Data Fusion based on Local Deviation of Wavelet Packet Transform," *Proceedings of the 2005 International Symposium on Autonomous Decentralized Systems (ISADS 2005)*, pp. 372-377, Chengdu, China, Apr. 2005.
2. Ya Qiu, Jin Wu, Honglin Huang, Jian Liu, Jinwen Tian and Yimin Liu, "Optimal Image Thresholding via Intelligent Genetic Algorithm Based on Phase Correlation for Large-Scale Image," *2005 6th International symposium on Test and Measurement (ISTM 2005)*, vol. 7, pp. 6305-6308, Dalian, China, Jun. 2005.
3. Ya Qiu, Jin Wu, Honglin Huang, Huaiyu Wu, Jian Liu and Jinwen Tian, "Multi-sensor image data fusion based on pixel-level weights of wavelet and the PCA transform," *Proceedings of the 2005 International Conference on*

-
- Mechatronics and Automation (ICMA 2005)*, pp. 653-658, Niagara Falls, Ontario, Canada, Jul. 2005.
4. Jin Wu, Honglin Huang, Ya Qiu, Huaiyu Wu, Jian Liu and Jinwen Tian, "Remote Sensing Image Fusion Based on Average Gradient of Wavelet Transform," *Proceedings of the 2005 International Conference on Mechatronics and Automation (ICMA 2005)*, pp. 1817-1822, Niagara Falls, Ontario, Canada, Jul. 2005.
 5. Jin Wu, Jinwen Tian, Jian Liu and Honglin Huang, "Multi-scale image data fusion based on local deviation of wavelet transform," *Proceedings of the 2004 International Conference on Intelligent Mechatronics and Automation (ICIMA 2004)*, pp. 677-680, UESTC, Chengdu, China, Aug. 2004.
 6. Jin Wu, Honglin Huang, Jinwen Tian and Jian Liu, "Remote sensing image data fusion based on IHS and local deviation of wavelet transformation," *Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics (ROBIO 2004)*, pp. 564-568, Shenyang, China, Aug. 2004.
 7. Jin Wu, Ya Qiu, Honglin Huang, Jian Liu and Jinwen Tian, "A Robust Hash Used In the Application of Digital Image Signature," *Proceedings of the 2004 International symposium on Distributed Computing and Applications to Business, Engineering and Sciences (DCABES 2004)*, vol. II, pp.982-985, Wuhan, China, Sept. 2004.

Bibliography

- [1] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, 41(12):3397-3415, 1993.
- [2] S. S. Chen, D. L. Donoho and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, 20(1):33-61 (1998).
- [3] S. S. Chen, D. L. Donoho and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, 43(1):129-159, 2001.
- [4] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, 24:227-234, 1995.
- [5] A. Feuer and A. Nemirovsky, "On sparse representation in pairs of bases," *IEEE Trans. on Information Theory*, 49:1579-1581, Jun. 2003.
- [6] J. J. Fuchs, "On sparse representations in arbitrary redundant bases," *IEEE Trans. on Information Theory*, 50:1341-1344, 2004.
- [7] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *IEEE Trans. on Information Theory*, 49(12):3320-3325, 2003.
- [8] R. Chartrand, "Exact reconstructions of sparse signals via nonconvex minimization," *IEEE Signal Proc. Letters*, 14(10):707-710, 2007.

-
- [9] A. M. Bruckstein, D. L. Donoho and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, 51(1):34-81, Feb. 2009.
 - [10] S. Mallat, *A wavelet Tour of Signal Processing*, Academic-Press, 1998.
 - [11] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*, Englewood Cliffs, NJ: Prentice Hall, 1995.
 - [12] A. N. Akansu and R. A. Haddad, *Multiresolution Signal Decomposition: Transform, Subband and Wavelets*, 2nd edition, Norwell, MA: Kluwer Academic, 2002.
 - [13] M. N. Do and M. Vetterli, "The finite ridgelet transform for image representation," *IEEE Trans. on Image Processing*, 12(1):16–28, 2003.
 - [14] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. on Image Processing*, 14(12):2091–2106, Dec. 2005.
 - [15] R. Eslami and H. Radha, "The contourlet transform for image de-noising using cycle spinning," in *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, pp. 1982–1986, Nov. 2003.
 - [16] S. Mallat and E. Le Pennec, "Sparse geometric image representation with bandelets," *IEEE Trans. on Image Processing*, 14(4):423-438, 2005.

-
- [17] S. Mallat and E. Le Pennec, “Bandelet image approximation and compression,” *SIAM Journal of Multiscale Modeling and Simulation*, 4(3):992–1039, 2005.
 - [18] E. J. Candès and D. L. Donoho, “Recovering edges in ill-posed inverse problems: optimality of curvelet frames,” *Annals of Statistics*, 30(3):784–842, 2000.
 - [19] E. J. Candès and D. L. Donoho, “New tight frames of curvelets and optimal representations of objects with piecewise-C² singularities,” *Comm. Pure Appl. Math.*, 57:219-266, 2002.
 - [20] D. L. Donoho, “De-noising by soft thresholding,” *IEEE Trans. on Information Theory*, 41(3):613–627, 1995.
 - [21] A. K. Fletcher, S. Rangan, V. K. Goyal and K. Ramchandran, “Denoising by sparse approximation: error bounds based on rate-distortion theory,” *EURASIP Journal on Applied Signal Processing*, Paper No. 26318, 2006.
 - [22] A. K. Fletcher, S. Rangan, V. K. Goyal and K. Ramchandran, “Analysis of denoising by sparse approximation with random frame asymptotics,” *IEEE Int. Symp. on Inform. Theory*, 2005.
 - [23] M. Elad, B. Matalon and M. Zibulevsky, “Image denoising with shrinkage and redundant representations,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, NY, Jun. 17-22, 2006.

-
- [24] J. -L. Starck, E. J. Candès and D. L. Donoho, “The curvelet transform for image denoising,” *IEEE Trans. on Image Processing*, 11:670–684, 2002.
 - [25] M. Protter, I. Yavneh and M. Elad, “Closed-form MMSE for denoising signals under sparse-representation modeling,” *The IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, Eilat Israel, Dec. 3-5, 2008.
 - [26] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. on Image Processing*, 15(12):3736–3745, Dec. 2006.
 - [27] O. G. Guleryuz, “Weighted averaging for denoising with overcomplete dictionaries,” *IEEE Trans. on Image Processing*, 16:30203034, 2007.
 - [28] M. Protter and M. Elad, “Image sequence denoising via sparse and redundant representations,” *IEEE Trans. on Image Processing*, 18(1):27–36, Jan. 2009.
 - [29] J. Mairal, G. Sapiro and M. Elad, “Learning multiscale sparse representations for image and video restoration,” *SIAM Multiscale Modeling and Simulation*, 7(1):214–241, Apr. 2008.
 - [30] M. A. Figueiredo and R. D. Nowak, “An EM algorithm for wavelet-based image restoration,” *IEEE Trans. Image Processing*, 12(8):906–916, 2003.
 - [31] K. Dabov, A. Foi and K. Egiazarian, “Image restoration by sparse 3D transform domain collaborative filtering,” *Proc. SPIE Electronic Imaging '08*, no. 6812-07, San Jose, California, USA, Jan. 2008.

-
- [32] J. Mairal, M. Elad and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. on Image Processing*, 17(1):53–69, Jan. 2008.
 - [33] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, "Non-local sparse models for image restoration," *International Conference on Computer Vision (ICCV)*, Tokyo, Japan, 2009.
 - [34] J. Ma, F. -X. Le Dimet, "Deblurring from highly incomplete measurements for remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, 47 (3):792-802, Mar. 2009.
 - [35] M. J. Fadili, J. -L. Starck and F. Murtagh, "Inpainting and zooming using sparse representations," *The Computer Journal*, 52(1):64–79, 2009.
 - [36] M. Elad, J. -L. Starck, P. Querre and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Journal on Applied and Comp. Harmonic Analysis*, 19:340–358, 2005.
 - [37] M. Bertalmio, L. Vese, G. Sapiro and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Trans. on Image Processing*, 12(8):882–889, Aug. 2003.
 - [38] M. J. Fadili and J. -L. Starck, "Sparse representation-based image deconvolution by iterative thresholding," *Astronomical Data Analysis ADA'06*, Marseille, France, Sep. 2006.

-
- [39] J. -L. Starck, M. Nguyen and F. Murtagh, “Wavelets and curvelets for image deconvolution: A combined approach,” *Signal Processing*, 83(10):2279–2283, 2003.
 - [40] C. Vonesch, S. Ramani and M. Unser, “Recursive risk estimation for non-linear image deconvolution with a wavelet-domain sparsity constraint,” the 15th International Conference on Image Processing, (ICIP), pp.665–668, 12-15 Oct. 2008.
 - [41] J. Yang, J. Wright, T. Huang and Y. Ma, “Image super-resolution as sparse representation of raw image patches,” *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2008.
 - [42] J. Yang, J. Wright, T. Huang and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Trans. on Image Processing*, 19(11): 2861-2873 Nov. 2010.
 - [43] T. Hazan, S. Polak and A. Shashua, “Sparse image coding using a 3D nonnegative tensor factorization,” in *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV)*, Beijing, China, pp. 17–21, Oct. 2005.
 - [44] L. Peotta, L. Granai and P. Vandergheynst, “Image compression using an edge adapted redundant dictionary and wavelets,” *Signal Processing*, 86:444–456, 2006.

-
- [45] A. J. Ferreira and M. A. T. Figueiredo, "On the use of independent component analysis for image compression," *Signal Processing: Image Communication*, 21:378–389, 2006.
 - [46] A. Said and W. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. on Circuits Systems for Video Technology*, 6(3):243–250, Jun. 1996.
 - [47] M. Zibulevsky and B. A. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural Computation*, 13(4):863–882, 2001.
 - [48] Y. Li, A. Cichocki and S. -i. Amari, "Analysis of sparse representation and blind source separation," *Neural Computation*, 16(6):1193–1234, 2004.
 - [49] J. Bobin, J. -L. Starck, M. J. Fadili and Y. Moudden, "Sparsity and morphological diversity in blind source separation," *IEEE Trans. on Image Processing*, 16(11):2662–2674, 2007.
 - [50] J. Bobin, Y. Moudden, J. -L. Starck and M. Elad, "Morphological diversity and source separation," *IEEE Signal Processing Letters*, 13(7):409–412, Jul. 2006.
 - [51] J. -L. Starck, M. Elad and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Trans. on Image Processing*, 14(10):1570–1582, 2005.

-
- [52] J. Mairal, M. Leordeanu, F. Bach, M. Hebert and J. Ponce, “Discriminative sparse image models for class-specific edge detection and image interpretation,” European Conference on Computer Vision (ECCV), Marseille, France, 2008.
 - [53] G. Peyré, “Sparse modeling of textures,” *Journal of Mathematical Imaging and Vision*, 34(1):17-31, May 2009.
 - [54] E. Candès, J. Romberg and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
 - [55] E. Candès, “Compressive sampling,” in *Proc. Int. Congress of Mathematicians*, vol. 3, pp. 1433-1452, Madrid, Spain, Aug. 2006.
 - [56] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
 - [57] Y. Tsaig and D. L. Donoho, “Extensions of compressed sensing,” *Signal Processing*, vol. 86, no. 3, pp. 549–571, Mar. 2006.
 - [58] G. Davis, S. Mallat and Z. Zhang, “Adaptive time-frequency decompositions,” *Optical-Engineering*, 33(7):2183-91, 1994.
 - [59] G. Davis, S. Mallat and M. Avellaneda, “Adaptive greedy approximations,” *Journal of Constructive Approximation*, 13:57-98, 1997.

-
- [60] J. A. Tropp, "Greed is good: Algorithmic results for Sparse approximation," IEEE Trans. On Information Theory, 50(10):2231-2242, Oct. 2004.
- [61] D. L. Donoho and M. Elad, "Optimally sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization," Proc. of the National Academy of Sciences, 100(5): 2197-2202, 2003.
- [62] I. Daubechies, M. Defrise and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," Communications on Pure and Applied Mathematics, 57(11):1413–1457, Nov. 2004.
- [63] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," Journal of Fourier Analysis and Applications, 14(5):629–654, 2008.
- [64] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," IEEE Signal Processing Magazine, vol. 25, no. 2, pp. 83-91, 2008.
- [65] M. B. Wakin, J. N. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly and R. G. Baraniuk, "An architecture for compressive imaging," in Proc. of the Int. Conf. on Image Processing (ICIP), pp. 1273-1276, Atlanta, Georgia, Oct. 2006.

-
- [66] M. Lustig, D. Donoho and J. M. Pauly, "Sparse MRI: the application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, 58 (6) pp. 1182-1195, Dec. 2007.
- [67] H. Jung, K. H. Sung, K. S. Nayak, E. Y. Kim and J. C. Ye, "k-t FOCUSS: a general compressed sensing framework for high resolution dynamic MRI," *Magnetic Resonance in Medicine*, vol. 61, pp. 103-116, Jan. 2009.
- [68] Y. -C. Kim, S. S. Narayanan and K. S. Nayak, "Accelerated three-dimensional upper airway MRI using compressed sensing," *Magnetic Resonance in Medicine*, 61: 1434-1440, 2009.
- [69] J. Laska, S. Kirolos, M. Duarte, T. Ragheb, R. Baraniuk and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, pp. 1959-1962, New Orleans, Louisiana, Jun. 2007.
- [70] M. Mishali and Y. C. Eldar, "Blind multi-band signal reconstruction: compressed sensing for analog signals," *IEEE Trans. on Signal Processing*, 57(30), pp. 993-1009, Mar. 2009.
- [71] M. Mishali and Y. C. Eldar, "From theory to practice: sub-Nyquist sampling of sparse wideband analog signals," *IEEE J. of Sel. Top. on Signal Processing*, 4(2), pp. 375-391, Apr. 2010.

-
- [72] A. Gholami and H. R. Siahkoobi, "Regularization of linear and non-linear geophysical ill-posed problems with joint sparsity constraints," *Geophysical J. Int.*, Vol. 180 (2), pp. 871-882, Feb. 2010.
- [73] M. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE Trans. on Signal Processing*, Vol. 57(6), pp. 2275-2284, 2009.
- [74] J. H. G. Ender, "On compressive sensing applied to radar," *Signal Processing*, vol. 90(5), pp. 1402-1414, May 2010.
- [75] X. X. Zhu and R. Bamler, "Tomographic SAR inversion by L_1 -norm regularization - the compressive sensing approach," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 48(10), pp. 3839-3846, 2010.
- [76] G. Tauböck, F. Hlawatsch, D. Eiwen and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: leakage effects and sparsity-enhancing processing," *IEEE J. Sel. Top. on Signal Processing*, vol. 4(2), pp. 255-271, Apr. 2010.
- [77] C. R. Berger, S. Zhou, J. C. Preisig and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: from subspace methods to compressed sensing," *IEEE Trans. on Signal Processing*, vol. 58(3), pp. 1708-1721, Mar. 2010.
- [78] C. R. Berger, Z. Wang, J. Huang and S. Zhou, "Application of compressive sensing to sparse channel estimation," *IEEE Communications Magazine*, vol. 48(11), pp. 164-174, Nov. 2010.

-
- [79] E. J. Candès and T. Tao, “Decoding by linear programming,” *IEEE Trans. on Information Theory*, vol. 51(12), pp. 4203–4215, Dec. 2005.
 - [80] M. Elad, “Optimized Projections for Compressed Sensing,” *IEEE Trans. on Signal Processing*, 55(12), pp. 5695-5702, Dec, 2007.
 - [81] M. Elad, *Sparse and Redundant Representations: from theory to applications in signal and image processing*, Springer, 2010.
 - [82] R. A. Devore and V. Temlyakov, “Some Remarks on Greedy Algorithms,” *Advances in Computational Mathematics*, 5:173-187, 1996.
 - [83] R. Gribonval and P. Vandergheynst, “On the exponential convergence of Matching Pursuits in quasi-incoherent dictionaries,” *IEEE Trans. Information Theory*, 52(1):255-261, 2006.
 - [84] V. N. Temlyakov, “Greedy algorithms and M-term approximation with regard to redundant dictionaries,” *J. of Approximation Theory*, 98:117-145, 1999.
 - [85] V. N. Temlyakov, “Weak greedy algorithms,” *Advances in Computational Mathematics*, 12:213-227, 2000.
 - [86] C. Couvreur and Y. Bresler, “On the optimality of the backward greedy algorithm for the subset selection problem,” *SIAM Journal on Matrix Analysis and Applications*, 21(3):797-808, 2000.

-
- [87] J. A. Tropp and A. A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Information Theory*, vol. 53(12), pp. 4655-4666, Dec. 2007.
- [88] P. Schniter, L. C. Potter and J. Ziniel, "Fast Bayesian matching pursuit," *Proc. Workshop on Information Theory and Applications (ITA)*, (La Jolla, CA), Jan. 2008.
- [89] D. L. Donoho, Y. Tsaig, I. Drori and J. -L. Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," Technical Report, Stanford University, 2006, available online: <http://www.dsp.ece.rice.edu/cs/>.
- [90] D. Needell and R. Vershynin, "Uniform Uncertainty Principle and Signal Recoery via Regularized Orthogonal Matching Pursuit," *Found. Comput. Math.*, vol. 9, pp. 317-334, 2009.
- [91] C. La and M. N. Do, "Signal reconstruction using sparse tree representations," in *Proc. of the SPIE*, vol.5914, pp. 59140W 1-11, Bellingham, WA, 2005.
- [92] W. Dai and O. Milenkovic, "Subspace Pursuit for Compressive Sensing Signal Reconstruction," *IEEE Trans. Inform. Theory*, vol.55, pp. 2230-2249, May, 2009.
- [93] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied Computational Harmonic Analysis*, 26:301-321, May 2009.

-
- [94] D. L. Donoho and J. Tanner, “Sparse nonnegative solutions of underdetermined linear equations by linear programming,” *Proceedings of the National Academy of Sciences*, 102(27): 9446-9451, Jul. 2005.
- [95] D. L. Donoho and M. Elad, “On the stability of the basis pursuit in the presence of noise,” *Signal Processing*, 86(3):511-532, Mar. 2006.
- [96] J. Cai, S. Osher and Z. Shen, “Convergence of the linearized Bregman iteration for ℓ_1 -norm minimization,” *Mathematics of Computation*, 78: 2127–2136, 2009.
- [97] E. J. Candès and J. Romberg, “Practical signal recovery from random projections,” *Wavelet Applications in Signal and Image Processing XI, Proc. SPIE Conf.*, vol. 5914, 2005.
- [98] D. L. Donoho, “For most large underdetermined systems of linear equations, the minimal ℓ_1 -norm solution is also the sparsest solution,” *Communications on Pure and Applied Mathematics*, 59(6):797–829, Jun. 2006.
- [99] I. F. Gorodnitsky and B. D. Rao, “Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimization norm algorithm,” *IEEE Trans. On Signal Processing*, 45(3):600-616, 1997.
- [100] E. J. Candès and T. Tao, “The Dantzig selector: Statistical estimation when p is much larger than n ,” *Ann. Statist.*, vol. 35(6): 2313–2351, Jun. 2007.

-
- [101] P. J. Bickel, Y. Ritov and A. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *Ann. Statist.*, vol. 37(4), pp. 1705-1732, 2009.
- [102] G. M. James, P. Radchenko and J. Lv, "DASSO: connections between the Dantzig selector and lasso," *Journal of the Royal Statist. Soc. B*, 71(1):127–142, 2009.
- [103] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. on Information Theory*, 47(7):2845-2862, 1999.
- [104] M. Elad and A. M. Bruckstein, "A generalized uncertainty principle and sparse representation in pairs of basis," *IEEE Trans. on Information Theory*, 48:2558-2567, 2002.
- [105] E. Candès and J. Romberg, "Quantitative robust uncertainty principles and optimally sparse decompositions," *Foundations of Computational Mathematics*, vol.6 (2), pp. 227-254, 2006.
- [106] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp.969-985, Jun. 2007.
- [107] E. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies," *IEEE Trans. Information Theory*, vol. 52(12), pp. 5406-5425, Dec. 2004.
- [108] M. Fornasier and H. Rauhut, "Iterative thresholding algorithms," *Applied and Computational Harmonic Analysis*, vol. 25(2), pp. 187-208, Sep. 2008.

-
- [109] K. Bredies and D. A. Lorenz, “Iterated hard shrinkage for minimization problems with sparsity constraints,” *SIAM J. on Scientific Computing*, vol. 30(2), pp. 657-683, Mar. 2008.
 - [110] K. Schnass and P. Vandergheynst, “Average performance analysis for thresholding,” *IEEE Signal Processing Letters*, 14(11): 828–831, Nov. 2007.
 - [111] E. Candès, J. Romberg and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Communications on Pure and Applied Mathematics*, vol. 59(8), pp. 1207-1223, Aug. 2006.
 - [112] E. Candès and M. Wakin, “An introduction to compressive sampling,” *IEEE Signal Processing Magazine*, vol.25 (2), pp. 21-30, Mar. 2008.
 - [113] S. J. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinevsky, “An interior-point method for large-scale ℓ_1 -regularized least squares,” *IEEE J. on Selected Topics in Signal Processing*, vol. 1(4), pp. 606-617, Dec. 2007.
 - [114] T. Blumensath, M. Yaghoobi and M. Davies, “Iterative hard thresholding and ℓ_0 regularization,” in *Proc. of the 2007 IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. 3, pp. 877-880, Apr. 2007.
 - [115] R. DeVore, “Deterministic Constructions of Compressed Sensing Matrices,” *J. of Complexity* 23, pp.918-925, Aug, 2007.

-
- [116] S. Gurevich, R. Hadani and N. Sochen, "On some deterministic dictionaries supporting sparsity," Special issue on sparsity, the Journal of Fourier Analysis and Applications, Vol. 14, 859-876, Dec. 2008.
- [117] R. Calderbank, S. Howard and S. Jafarpour, "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property," IEEE J. of Selected Topics in Signal Processing, vol. 4(2), pp.358-374, Apr. 2010.
- [118] I. E. Nesterov, A. Nemirovskii and Y. Nesterov, Interior-point Polynomial Algorithms in Convex Programming, Philadelphia, PA: SIAM, 1994.
- [119] T. T. Do, L. Gan, N. Nguyen and T. D. Tran, "Sparsity adaptive matching pursuit algorithm for practical compressed sensing," 42nd Asilomar Conf. on Signals, Systems and Computers (ACSSC), pp.581-587, Oct. 2008.
- [120] SparseLab, <http://dsp.rice.edu/cs> [available online].
- [121] Å. Björck, Numerical Methods for Least Squares Problems, SIAM, 1996.
- [122] M. A. T. Figueiredo, R. D. Nowak and S. J. Wright, "Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems," IEEE J. of selected topics in signal processing: Special Issue on Convex Optimization Methods for Signal Processing, vol.1, no.4, pp.586-597, Dec. 2007.

-
- [123] h.264 JM Reference Software Version 10.2, available online:
http://iphome.hhi.de/suehring/tml/download/old_jm/jm10.2.zip.
- [124] A. Vetro, T. Haga, K. Sumi and H. Sun, "Object-based coding for long-term archive of surveillance video," in IEEE International Conf. on Multimedia and Expo, July 2003.
- [125] C. Kim and J-N. Hwang, "Object-based video abstraction for video surveillance system," IEEE Trans. Circuits and Systems for Video Technology, vol.12, no. 12, pp.1128-1138, Dec. 2002.
- [126] R. V. Babu and A. Makur, "Object-based surveillance video compression using foreground motion compensation," in Proc. of the 9th Conf. Control, Automation, Robotics and vision (2006), pp.458-463, Singapore, Dec. 2006.
- [127] A. Elgammal, R. Duraiswami, D. Harwood and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," Proceedings of the IEEE, vol.90, no.7, pp.1151-1163, Jul. 2002.
- [128] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in IEEE International Conf. on Computer Vision and Pattern Recognition, pp.246-252, 1999.
- [129] H. Freeman, "On the encoding of arbitrary geometric configurations," IRE Trans. Electronic Computers, vol.EC-10, pp. 260-268, June 1961.

-
- [130] "Caviar: Context aware vision using image-based active recognition," Available online: <http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>.
- [131] D. C. Youla, "Mathematical Theory of Image Restoration by the Method of Convex Projection," in Image Recovery-Theory and Application, New York Academic, 1987.
- [132] D. C. Youla and H. Webb, "Image Restoration by the Method of Convex Projections: Part I-Theory," IEEE Tran. Medical Imaging, vol. 1(2), pp. 81-94, Oct. 1982.
- [133] M. I. Sezan and H. Stark, "Image Restoration by the Method of Convex Projections: Part 2-Applications and Numerical Results," IEEE Tran. Medical Imaging, vol. 1(2), pp. 95-101, Oct. 1982.
- [134] J. J. Zou and H. Yan, "A Deblocking Method for BDCT Compressed Images Based on Adaptive Projections," IEEE Tran. on Circuits and Systems for Video Technology, vol.15, no.3, Mar. 2005.
- [135] H. Paek, R. C. Kim and S. U. Lee, "On the POCS-based Postprocessing Technique to Reduce the Blocking Artifacts in Transform Coded Images," IEEE Tran. on Circuits and Systems for Video Technology, vol.8, no.3, Jun. 1998.
- [136] V. K. Goyal, A. K. Fletcher and S. Rangan, "Compressive sampling and lossy compression," IEEE Signal Processing Magazine, vol. 25(2), pp. 48-56, Mar. 2008.