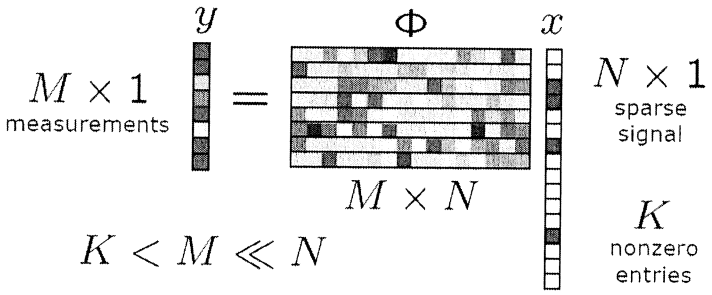
CHALMERS UNIVERSITY OF GOTHENBURG

Dear Heng!

This is very good indeed.

I) Some of your images are not very clear. It does not tell very much how the methods are improving. Can you, possibly, put sharper images?

MASTER'S THESIS



Compressed sensing – with applications to medical imaging

N/A

HENG YANG

II) You need to improve the quality of the presentation. To this end I put some suggestion in the text.

Department of Mathematical sciences
Division of Mathematics
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2011

III) There is a minimum amount of Mathematical motivation / arguments. Please enrich the presentation by adding adequate

Thesis for the Degree of Master of Science

Compressed sensing – with applications to medical imaging

Heng Yang

Department of Mathematical Sciences

Chalmers University of Technology and Gothenburg University

SE-412 96 Göteborg, Sweden

Göteborg, June 2011

Mathematics Sciences
Göteborg 2011

Abstract

Compressed sensing is a new approach for acquiring signals. It captures and represents signals and images at a rate significantly below Nyquist rate. In certain areas like magnetic resonance imaging (MRI), it is urgent to reduce the time of the patients' exposure in the electromagnetic radiation. Compressed sensing breaks the canonical rules and effectively reduces the sampling rate without losing the essential information, so it has a wide application in medical imaging. In this project, different recovery strategies will be explored to investigate the performance of algorithms on different MRI images.

Key words: Compressed sensing, Magnetic resonance imaging, Recovery strategy

Extend your key words containing the three methods, PSNR ...

Acknowledgments

First I would like to give my sincerest gratitude to my supervisor Prof. Morten Nielsen in Aalborg University. Thank you for leading me to the world of compressed sensing and always be patient to answer my questions. His well organized working process and scientific attitude will be a great treasure for my whole life. In addition, I want to thank my examiner Mohammad Asadzadeh. Thank you for being the bridge between Chalmers and Aalborg University. Without your constant support, I wouldn't finish the thesis in time.

Special thanks for my family and the people who love me and support me always.

Contents

Notation 2
1 Introduction 3
1.1 Background 3
1.1.1 The drawback of Nyquist sampling 3
1.1.2 The sparsity of signals 3
1.1.3 The inefficiencies of conventional data transform 4
1.1.4 The introduction of compressed sensing 4
1.2 Purpose 5
1.3 Outline 5
2 How to do reconstruction 6
2.1 Restricted isometry property (RIP) 6
2.2 The idea of Reconstruction 6
3 The reconstruction algorithms 8
3.1 OMP algorithm 8
3.1.1 Description 8
3.1.2 Experiments 9
3.1.3 Conclusion 11
3.2 CoSaMP algorithm 12
3.2.1 Description 12
3.2.2 Experiments 13
3.2.3 Conclusion 16
3.3 Model-based compressive sensing 16
3.3.1 Description 16
3.3.2 Experiments 21
3.3.3 Conclusion 25
4 Experiments on MRI Images 26
5 Conclusion 34
Reference 35

Notation

$\mathbb{R}$	Real numbers
$\mathbf{x}$	Real valued, finite length signal with length N
$\|\mathbf{x}\|_p, 1 \leq p < \infty$	P-norm $\|\mathbf{x}\|_p = \left(\sum_{i=1}^{N} \|x_i\|^p\right)^{1/p}$
$\|x\|_0$	Quasi-norm $\|x\|_0 = \|\mathrm{supp}(x)\|$
$\#T$, $\|T\|$	Cardinality of T
$\mathbf{\Phi}$	Measurement matrix with size $M \times N$
$\langle f, g \rangle$	Inner product
$\mathbf{A}^\dagger$	Pseudoinverse of full-rank matrix $\mathbf{A}$. $\mathbf{A}^\dagger = (\mathbf{A}^*\mathbf{A})^{-1}\mathbf{A}^*$
Λ	Index set

supp(x):= $\{j : x_j \neq 0\}$

$\sum_K := \{x \in C^N : \|x\|_0 \leq K\}$

1 Introduction

1.1 Background

1.1.1 The drawback of Nyquist sampling

According to the Shannon/Nyquist sampling theorem, in order to reconstruct a bandlimited signal perfectly the sampling rate should be at least two times that of the signal bandwidth [3]. To be more specific, let $x(t)$ represent a continuous-time signal and $X(f)$ be the continuous Fourier transform of the signal $x(t)$, we have:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}dt$$

The signal $x(t)$ is said to be bandlimited if there is a B, such that $X(f) = 0$ for all $|f| > B$. Figure 1.1 shows an example of a bandlimited signal $x(t)$. The quantity 2B is called the Nyquist rate. The sufficient condition for signal $x(t)$ to be perfectly reconstructed from an infinite sequence of samples is the sample rate f_s should be larger than 2B. If f_s is less than 2B, aliasing will be introduced after reconstruction.

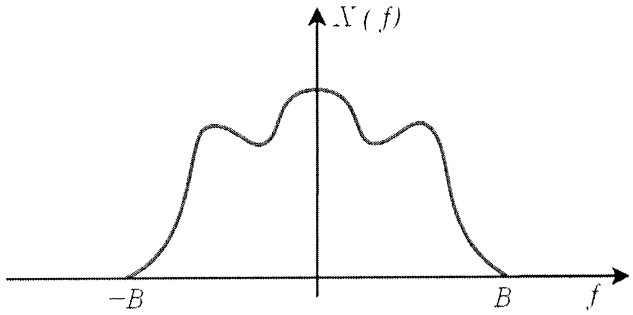


Figure 1.1: An example the Fourier transform of a bandlimited signal $x(t)$[3]

While in reality, this sampling rate is still so high that too many samples should be achieved. Especially in the medical imaging modality, we need to reduce the time of the patients' exposure in the electromagnetic radiation. So it is desirable to take as few samples as possible without losing essential information. It is interesting to notice that most signals in reality are sparse. When they are represented in some domain (such as the wavelet domain), they contain many coefficients close to or equal to zero. Compressed sensing acquires and reconstructs a signal applying the prior knowledge that it is sparse. It can capture and represents compressible signals at a rate significantly lower than Nyquist rate.

1.1.2 The sparsity of signals

Using mathematics to illustrate, we have a discrete-time signal $\mathbf{x}$ in $\mathbb{R}^N$ that can be represented in terms of an orthonormal basis of $N \times 1$ vectors $[\boldsymbol{\psi}_i]_{i=1}^N$ as follows:

$$\mathbf{x} = \sum_{i=1}^{N} s_i \Psi_i \tag{1}$$

where s_i is the coefficient sequence of $\mathbf{x}$. For simplification, we can write (1) in matrix form as $\mathbf{x} = \boldsymbol{\psi}\mathbf{s}$ (where $\mathbf{s}$ is the $N \times 1$ column vector and $\boldsymbol{\psi}$ is the $N \times N$ matrix with $\boldsymbol{\psi}_i$ as columns). Signal $\mathbf{x}$ has a K-sparse expansion if only K of the entries in $\mathbf{s}$ are non-zero and $(N - K)$ are zero. Real signals are often compressible which means the sequence of coefficients decays quickly. It means the large fraction of small coefficients can be thrown away without much perceptual loss.

1.1.3 The inefficiencies of conventional data transform

In traditional data acquisition, the first step is to acquire the full N-sample signal $\mathbf{x}$; then compute the coefficients $\{s_i\}$ via $\mathbf{s} = \boldsymbol{\psi}^T\mathbf{x}$ and only keep the K largest $\{s_i\}$ while discarding the others. The values and locations of the K largest $\{s_i\}$ should be encoded. This traditional signal acquisition processing divides the sampling and compression into two separate processes which samples a lot of unnecessary information. This inefficiency is more obvious when the number of samples N is large compared to K. Compressed sensing is a method to skip the sampling step by directly acquiring the compressed signal representation to overcome these inefficiencies.

1.1.4 The introduction of compressed sensing

In order to measure all the N coefficients of $\mathbf{x}$, we consider the $M \times 1$ $(M < N)$ column inner products $\mathbf{y}$ between $\mathbf{x}$ and collection of vectors $\{\boldsymbol{\Phi}_j\}_{j=1}^{M}$:

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{x} = \boldsymbol{\Phi}\boldsymbol{\psi}\mathbf{s} = \boldsymbol{\Theta}\mathbf{s} \tag{2}$$

where $\boldsymbol{\Theta} = \boldsymbol{\Phi}\boldsymbol{\psi}$ is an $M \times N$ matrix. $\boldsymbol{\Phi}$ is called an $M \times N$ measurement matrix with $\boldsymbol{\Phi}_j^T$ as rows. $\boldsymbol{\Phi}$ is fixed and does not depend on the signal $\mathbf{x}$, so this process is non-adaptive. This is a great point since if we get a robust result from a measurement matrix $\boldsymbol{\Phi}$, we can apply this measurement matrix $\boldsymbol{\Phi}$ to any kinds of signals without worrying about the stability. Figure 1.2 illustrates the process of compressed sensing.

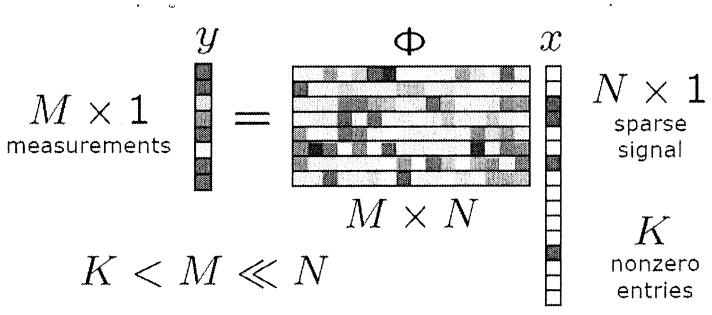


Figure 1.2: Compressed sensing measurement process [4]

The main task of this thesis is to investigate the algorithms about reconstructing the K- sparse signal $\mathbf{x} \in N \times 1$ by the given measurement vector $\mathbf{y} \in M \times 1$, with $M \ll N$ and $M = \mathcal{O}(K)$.

1.2 Purpose

The purpose of this project is to explore the compressed sensing strategy to reconstruct images stably and efficiently by using as few measurements as possible. Three reconstruction methods OMP, CoSaMP and Model-based algorithms will be illustrated and analyzed.

1.3 Outline

Chapter 1 gives a brief background of the thesis. Chapter 2 shows the idea of reconstruction and depicts the theory behind it. Chapter 3 emphasizes on the reconstruction algorithms of OMP, CoSaMP and Model-based, some experiments will be carried out as well. In Chapter 4, some real MRI images will be tested for further investigation by using different recovery strategies. Finally, a conclusion will be made in Chapter 5.

2 How to do reconstruction

2.1 Restricted isometry property (RIP)

The main task of encoding is to transform the $N \times 1$ signal $\mathbf{x}$ to the $M \times 1$ measurement $\mathbf{y}$ by using a proper measurement matrix $\mathbf{\Phi}$. The sampling matrix $\mathbf{\Phi}$ must map two different signals to two different sets of measurements, so all of the column submatrices (containing at most k columns) of $\mathbf{\Phi}$ should be well-conditioned.

Candès and Tao proposed a condition for the sampling matrix $\mathbf{\Phi}$. For all K-sparse vector $\mathbf{x}$, an $M \times N$ matrix $\mathbf{\Phi}$ has the K-restricted isometry property if

$$(1 - \delta_K)\|\mathbf{x}\|_2^2 \leq \|\mathbf{\Phi x}\|_2^2 \leq (1 + \delta_K)\|\mathbf{x}\|_2^2 \qquad (3)$$

When δ_K is less than 1, the inequalities (3) imply the all of the submatrices of $\mathbf{\Phi}$ with K columns are well-conditioned and close to an isometry. If $\delta_K \ll 1$, the sampling matrix $\mathbf{\Phi}$ has a large probability to reconstruct the (K/2)-sparse signal $\mathbf{x}$ stably.

The connection between RIP and CS is if δ_{2K} is sufficiently less than 1, all pairwise distances between K-Sparse signals must be well preserved in the measurement space which implies that $(1 - \delta_{2K})\|\mathbf{x_1} - \mathbf{x_2}\|_2^2 \leq \|\mathbf{\Phi x_1} - \mathbf{\Phi x_2}\|_2^2 \leq (1 + \delta_{2K})\|\mathbf{x_1} - \mathbf{x_2}\|_2^2$ holds for all K-sparse vectors $\mathbf{x_1}$ and $\mathbf{x_2}$. It can also be said that the sampling matrix $\mathbf{\Phi}$ should map two different K-sparse signals to different samples.

So as to invert the sampling process stably and get a K-sparse signal $\mathbf{x}$, we need to get a small restricted isometry constant δ_{2K}. However, it is computational difficult to check whether a matrix $\mathbf{\Phi}$ satisfies the inequality (3). Fortunately, many types of random matrices have a good restricted isometry behavior, and they satisfy the restricted isometry condition with high probability. One of the quintessential examples is Gaussian measurement matrix $\mathbf{\Phi}$, that the entries $\mathbf{\Phi}_j$ of $\mathbf{\Phi}$ are independent and identically distributed random variables from a Gaussian probability density function. An $M \times N$ IID Gaussian matrix has restricted isometry behavior with high probability if $M \geq c \cdot K \cdot \log(N/K)$ where c is a constant [1] [2]. This also means K-spare or compressible signals with length N can be recovered with M random Gaussian measurements.

In my project, I just pick a random matrix without checking its restricted isometry property. The random sampling matrix is regarded to have a good RIP behavior if the recovery signal is stable and approximately accurate.

2.2 The idea of Reconstruction

In order to achieve an optimal recovery algorithm, there are several requirements should be satisfied. First the algorithm should be stable. That means when the signals or the measurements are perturbed slightly by noise, recovery should still be approximately accurate. Then the algorithm should be fast if we want to apply it into practice. In addition, when acquiring linear measurements by using a specific method, these linear measurements can apply to all sparse signals. This is called the uniform guarantees. Finally, the algorithm should require as few measurements as possible.

Now we want to reconstruct a K-sparse signal $\mathbf{x} \in \mathbb{R}^N$ by the $M \times 1$ measurement vector $\mathbf{y} =$

$\mathbf{\Phi x}$. Since the measurement matrix $\mathbf{\Phi} \in \mathrm{M} \times \mathrm{N}$ and $\mathrm{M} < N$, the system (2) is underdetermined. Theoretically, there are infinitely many $\tilde{\mathbf{x}}$ that satisfy Eq. (2). However, in our case the additional assumption is that $\mathbf{x}$ is K-sparse, and then there is often a unique $\tilde{\mathbf{x}}$ that will suffice to recover $\mathbf{y}$. The best solution will be the sparsest vector that means it has the most zero coefficients. Consider the ℓ_0 -norm that counts the number of non-zeros entries, the reconstruction problem turns to be:

$$\mathbf{x} = \arg\min_{\tilde{\mathbf{x}}} \|\tilde{\mathbf{x}}\|_0, \text{ subject to } \mathbf{y} = \Phi\tilde{\mathbf{x}} \tag{4}$$

Unfortunately, the ℓ_0-minimization problem is NP-hard [25][26]. It is computationally intractable to solve Eq.(4) for any matrix $\mathbf{\Phi}$ and vector $\mathbf{y}$. So there are two families can be alternatively used to solve Eq.(4). One is the basic pursuit that is a convex relaxation leading to ℓ_1-norm minimization [18], the other is greedy pursuit [9] such as Orthogonal Matching Pursuit (OMP) [15], Stagewise Orthogonal Matching Pursuit (StOMP) [19], and Regularized Orthogonal Matching Pursuit (ROMP) [22][23].

ℓ_1-minimization approach

As we discussed in section 2.1, in most cases if the RIP holds, the ℓ_1-norm can exactly recover K-sparse signals and do a proper job to approximate the compressible signals with high probability using only $\mathrm{M} = \mathcal{O}(\mathrm{K} \cdot \log(\mathrm{N}/\mathrm{K}))$ IID Gaussian measurements. Then the Eq.(4) will change to be:

$$\mathbf{x} = \arg\min_{\tilde{\mathbf{x}}} \|\tilde{\mathbf{x}}\|_1, \text{ subject to } \mathbf{y} = \Phi\tilde{\mathbf{x}} \tag{5}$$

Eq. (5) is equivalent to the linear programming

$$\min \sum_{j=1}^{2N} v_j, \text{ subject to } \mathbf{v} \geq 0,\ \mathbf{y} = (\mathbf{\Phi}, -\mathbf{\Phi})\mathbf{v} \tag{6}$$

where $\mathbf{v} \in \mathbb{R}^{+2\mathrm{N}}$. The signal $\mathbf{x}$ is obtained from the solution $\mathbf{v}^*$ of (6) via $\mathbf{x} = (\mathbf{I}, -\mathbf{I})\mathbf{v}^*$. So the ℓ_1-norm minimization can be solved by linear programming problem. Interior-point methods [12], projected gradient methods [13] and iterative thresholding [14] can be used to solve the Eq. (6).
The ℓ_1-minimization approach can provide stability and uniform guarantees. But it doesn't have linear bound on the runtime, it is not optimally fast.

Greedy pursuit

Greedy pursuit is another approach to reconstruct the signal. It is an iterative signal recovery algorithm to calculate the support of the signal and it makes the locally optimal choice at each time to build up an approximation and repeats until the criterion is fulfilled. When we get the support S of the signal, the signal $\mathbf{x}$ can be reconstructed by $\mathbf{x} = (\mathbf{\Phi_s})^{\dagger}\mathbf{y}$, where $\mathbf{\Phi_s}$ is the measurement matrix $\mathbf{\Phi}$ with entries indexed by S and $(\mathbf{\Phi_s})^{\dagger}$ is the pseudoinverse of $\mathbf{\Phi}$. The pseudoinverse of a full-rank matrix $\mathbf{\Phi}$ is defined by the formula $\mathbf{\Phi}^{\dagger} = (\mathbf{\Phi}^*\mathbf{\Phi})^{-1}\mathbf{\Phi}^*$.
Greedy pursuit is extremely fast while it is not optimally stable and doesn't have uniform guarantees.

3 The reconstruction algorithms

3.1 OMP algorithm

3.1.1 Description

Orthogonal matching pursuit is a so called greedy algorithm for signal recovery. It was proposed by Mallat and Zhang [20] and analyzed by Gilbert and Tropp [9]. Suppose $\mathbf{x}$ is a K-sparse signal in $\mathbb{R}^N$, and let $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$ be a measurement matrix with columns $\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2 \dots, \boldsymbol{\varphi}_N$. Then the signal $\mathbf{x}$ can be represented by an M-dimensional measurement vector $\mathbf{y} = \mathbf{\Phi x}$. Since $\mathbf{x}$ has only K non-zero components, $\mathbf{y}$ can be regarded as a linear combination of K columns from $\mathbf{\Phi}$. So when we do the signal recovery, the most challenge part is to identify the location of the largest ideal signal $\mathbf{x}$. It is important to determine which columns of $\mathbf{\Phi}$ participate in the measurement vector $\mathbf{y}$. OMP applies the greedy algorithm to pick the columns of $\mathbf{\Phi}$ by finding the largest correlation between $\mathbf{\Phi}$ and the residual of $\mathbf{y}$. At each iteration, one coordinate for the support of the signal $\mathbf{x}$ is calculated. Hopefully after K iterations, the entire support of the signal $\mathbf{x}$ will be identified.

Tropp and Gilbert gave a proof about the weak uniform guarantees about the OMP [15]. They showed that OMP can correctly reconstruct the K-sparse signal $\mathbf{x}$ from its measurements $\mathbf{\Phi x}$ with probability exceeding $1 - 2\delta$ if $\mathbf{\Phi}$ is an $M \times N$ Gaussian measurement matrix with $M \geq C \cdot M \cdot \log(N/\delta)$. Here δ is a fixed constant between 0 and 0.36 and C is a constant. This guarantee is only for a fixed signal $\mathbf{x}$ not for all of signals. OMP may fail for some spares signals. It is also unknown whether OMP works for compressible signals rather than sparse signals or succeeds when samples contain noise.

The OMP algorithm has 4 major steps during each iteration:

(1) Find the index λ_i by choosing the largest correlation between $\{\boldsymbol{\varphi}_j\}_1^N$ and the residual of $\mathbf{y}$.

(2) Unite the newly chosen index λ_i with the index set $\Lambda_i = [\Lambda_{i-1}\ \lambda_i]$, and $\boldsymbol{\varphi}_{\lambda_i}$ with the matrix $\mathbf{\Phi_i} = [\mathbf{\Phi}_{i-1}\ \boldsymbol{\varphi}_{\lambda_i}]$. Here, $\mathbf{\Phi_0}$ is an empty set.

(3) Form the signal estimate $\mathbf{x_i}$ by using the least squares method which is to find the projection of $\mathbf{y}$ onto the range of $\mathbf{\Phi_i}$. It is easy to recognize that the residual $\mathbf{r_i} = \mathbf{y} - \mathbf{a_i}$ is always orthogonal to $\mathbf{\Phi_i}$.

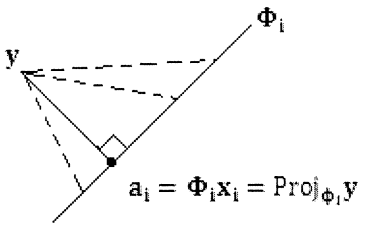


Figure 3.1: least square method

(4) Calculate the newly residual of $\mathbf{r_i}$ and then return to (1) if $i < K$.

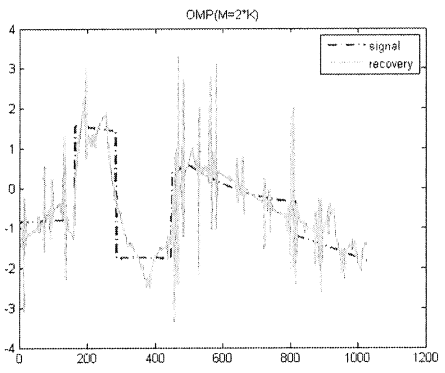
Once the support Λ of signal $\mathbf{x}$ is found, the approximation of signal $\tilde{\mathbf{x}}$ can be found by $\tilde{\mathbf{x}} = (\mathbf{\Phi}_\Lambda)^\dagger \mathbf{y}$.

Algorithm 1: OMP

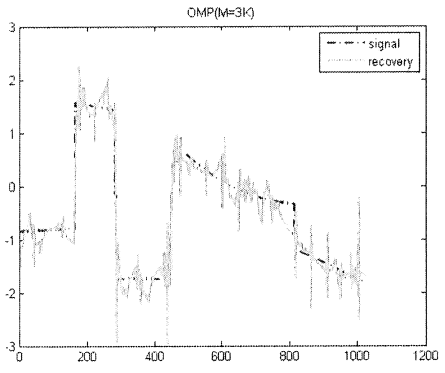
Input: Measurement matrix $\mathbf{\Phi}$, measurement $\mathbf{y}$, sparsity level K of the ideal signal $\mathbf{x}$ Output: index set Λ, measurement estimate $\mathbf{a_i}$, residual $\mathbf{r_i} = \mathbf{y} - \mathbf{a_i}$	
$\mathbf{r_0} = \mathbf{y}$, $\mathbf{\Lambda_0} = \emptyset$, $i = 0$	{Initialize}
while $i \leq K$ do	
1. $i = i + 1$	
2. $\lambda_i = \arg\max_{\{j=1,\ldots N\}} \lvert\langle \boldsymbol{r}_{i-1}, \boldsymbol{\varphi}_j \rangle\rvert$	{identify index of largest correlation}
3. $\Lambda_i = \Lambda_{i-1} \cup \{\lambda_i\}$	{Augment the index set}
4. $\mathbf{\Phi_i} = [\mathbf{\Phi_{i-1}} \quad \boldsymbol{\varphi}_{\lambda_i}]$	{Augment the matrix}
5. $\mathbf{x_i} = \arg\min_{\mathbf{x}} \lVert\mathbf{\Phi_i x} - \mathbf{y}\rVert_2$	{Signal estimate by least squares}
6. $\mathbf{a_i} = \mathbf{\Phi_i x_i}$, $\mathbf{r_i} = \mathbf{y} - \mathbf{a_i}$	{Update the new residual}
end while	

3.1.2 Experiments

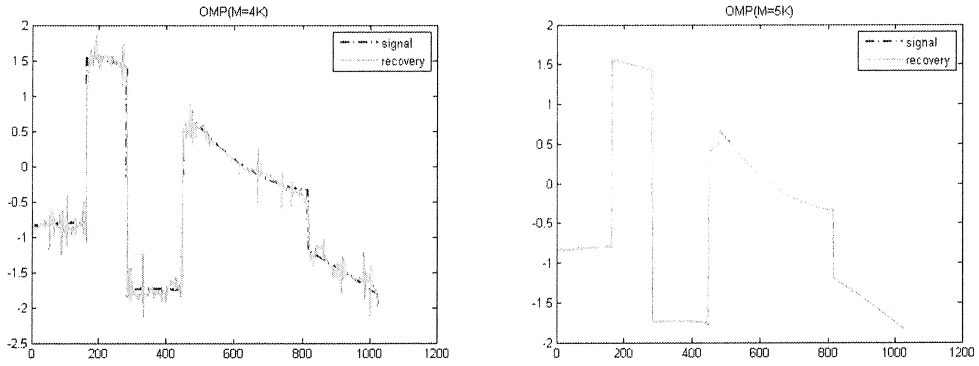
Figure 3.2 illustrates the results of a simulation study on the impact of the number of measurements M on the performance of OMP recovery. A piecewise polynomial signal of length N=1024 is generated by random. The signal in a piecewise polynomial wavelet basis is K-spare and K is equal to 42. Figure 3.2(a) to (d) show the performances of OMP recovery from M=2K, M=3K, M=4K and M=5K noise free measurements respectively. The figure shows that as the M growing, the recovery results are more and more accurate. When M=5K, the recovery and original signal almost overlap.



(a)



(b)



(c) (d)

Figure 3.2: Example performance of OMP signal recovery. The red dotted line is the orignal signal, the green solid line is the signal recovery. (a) Signal recovery from M=2K samples. (b) Signal recovery from M=3K samples. (c) Signal recovery from M=4K samples. (d) Signal recovery from M=5K samples.

Then let us study the behavior of OMP algorithm when the orignal signal is perturbed by noise. Here, the peak signal-to-noise ratio (shorted for PSNR) is used to measure the quality of reconstrucion. PSNR is a term of ratio between the maximum possible power of a signal and the power of corrupting noise. The PSNR is defined as:

$$PSNR = 10 \cdot \log_{10}(\frac{MAX_I^2}{MSE})$$

Here, I is a $m \times n$ signal and MAX_I is the maximum component of signal. MSE is the mean squared error and defined as:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

and K is an approximation of signal. Typical values for the PSNR are between 30 dB and 50 dB and acceptable values are considered to be about 20 dB to 25 dB. The higher PSNR is, the better recovery performance is. [11]

Since the measurements are generated though the multiplication of signal and measurement matrix $\mathbf{\Phi}$, the noise is also perturbed the measurements. Figrue 3.3 gives the PSNR of OMP under perturbed signal. The red pentagram line is the performance without noise. When M=4K, the PSNR is around 22 dB. But when M grows to 5K, the PSNR suddenly increases to approximately 81 dB. So for the noise free case, when M is bigger than 5K, the recovery performance is really good. But as noise gradually adding to the signal, even we have a large M, the reconstruction effect is still bad. The PSNR for $M \geq 6K$ with noise power level that is bigger than 0.2 is less than 20 dB.

It is similar when we add noise directly to the measurements. Figure 3.4 shows the PSNR result with noise measurements and noise power level is increased from 0 to 0.5 with increasing step 0.1. The quality of reconstruction is very poor if measurements are interfered by noise.

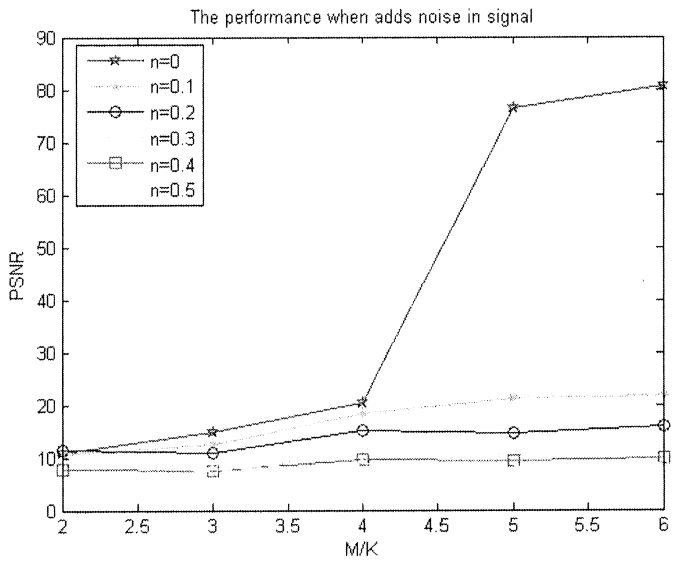


Figure 3.3: The stability tests of the OMP algorithm when random noise is added to the original signal. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

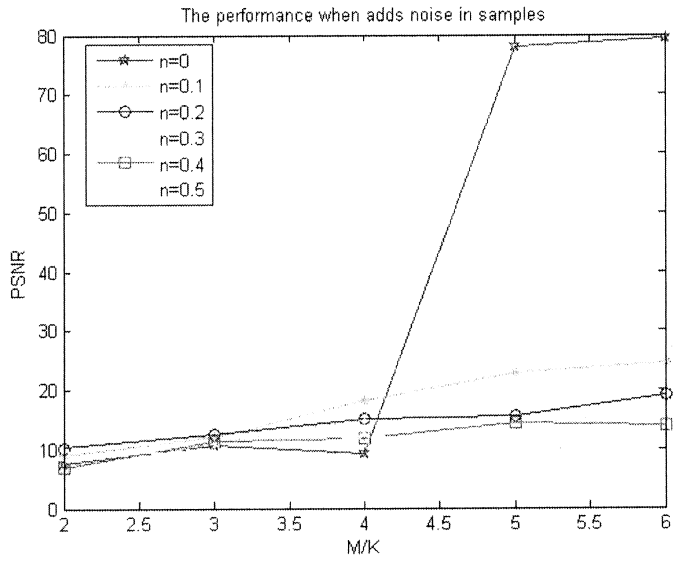


Figure 3.4: The stability tests of the OMP algorithm when we add noise to measurements. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

3.1.3 Conclusion

From the results Figure 3.3 and 3.4, it is easy to draw a conclusion that OMP can achieve a really ideal result without interuption of noise when M is big. But this algorithm is not stable. When the signals or the measurements are perturbed slightly by noise, recovery is not accurate any more.

3.2 CoSaMP algorithm

3.2.1 Description

CoSaMP, short for the Compressive Sampling Matching Pursuit, is a new reconstruction algorithm based on OMP (orthogonal matching pursuit). CoSaMP was first put forth by Needell and Tropp in 2009 [24].

The same as the OMP, identify the location of the K largest ideal signal $\mathbf{x}$ is the most important target. According to the restricted isometry property, by giving a sampling matrix $\mathbf{\Phi}$ with the restricted isometry constant $\delta_K \ll 1$, the ℓ_2-norm of the largest K entries of vector $\mathbf{u} = \mathbf{\Phi}^*\mathbf{\Phi}\mathbf{x}$ is close to the ℓ_2-norm of the largest K entries of the K-sparse signal $\mathbf{x}$. So $\mathbf{u}$ ~~that is equal to $\mathbf{\Phi}^*\mathbf{\Phi}\mathbf{x}$~~ can be called as a proxy of signal $\mathbf{x}$. The proxy $\mathbf{u}$ can be obtained by applying the matrix $\mathbf{\Phi}^*$ to the measurement $\mathbf{y}$. In order to identify the location of the largest K components of $\mathbf{x}$, it is enough to find out the location of the largest K components of the proxy $\mathbf{u}$.

At each iteration, the algorithm first selects the largest 2K components of the signal proxy $\mathbf{u}$ and adds the index of these components to the support set. Next using the least squares, we can get a signal estimation $\mathbf{b}$. The sparse signal $\mathbf{x}$ can be obtained by keeping only the largest K components of the estimation $\mathbf{b}$ to make it sparse. This is called pruning.

Needell and Tropp established the following result that for an arbitrary signal $\mathbf{x}$ with noise samples $\mathbf{y} = \mathbf{\Phi}\mathbf{x} + \mathbf{e}$, CoSaMP produces a 2K-sparse signal approximation $\hat{\mathbf{x}}$ that satisfies

$$\|\hat{\mathbf{x}} - \mathbf{x}\|_2 \leq C(\|\mathbf{e}\|_2 + \frac{\|\mathbf{x}-\mathbf{x}_k\|_1}{\sqrt{K}})$$

where $\mathbf{x}_k$ is a best K-sparse approximation of $\mathbf{x}$, $\mathbf{\Phi}$ is an $M \times N$ sampling matrix with restricted isometry constant $\delta_{2K} \leq c$ and c, C refer to positive constants.

This result illustrates without the interrupted of noise, CoSaMP can recover an arbitrary signal with high precision. What's more, the performance of recovery reduces gracefully if we add noise in the samples. The ℓ_2-norm of 2K-sparse signal approximation $\hat{\mathbf{x}}$ that is produced by this algorithm is comparable with the ℓ_1-norm of the best K-sparse approximation $\mathbf{x}_k$.

The CoSaMP has 5 major steps during each iteration:

(1) Find the proxy of the current samples' residual.
(2) Locate the largest 2K components of the proxy and unite with the index of the current signal approximation.
(3) Estimate the signal on the merged set of components by using the least squares.
(4) Choose the K largest components as the signal approximation to prune the signal estimation
(5) Update the samples' residual.

Algorithm 2: CoSaMP

Input: Measurement matrix $\mathbf{\Phi}$, measurement $\mathbf{y}$, sparsity level K of the ideal signal $\mathbf{x}$
Output: K-sparse estimate $\hat{\mathbf{x}}$, index set Λ, measurement estimate $\mathbf{a_m}$, residual $\mathbf{r} = \mathbf{y} - \mathbf{a_m}$

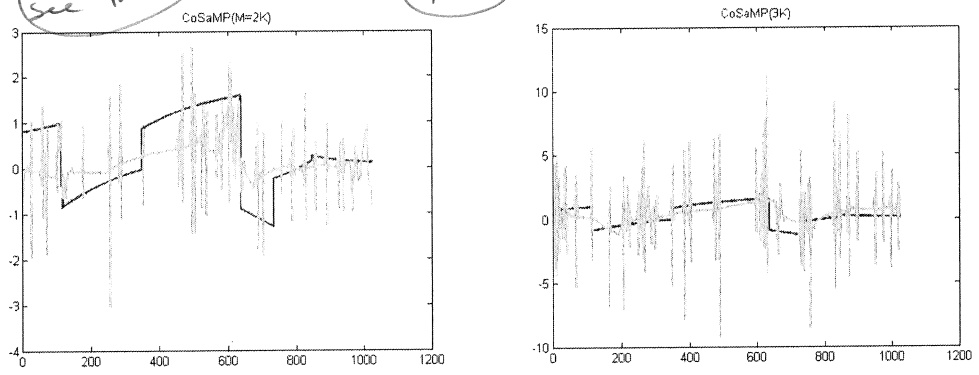
$\mathbf{x}_0 = 0,\ \mathbf{r_0} = \mathbf{y},\ \Lambda_0 = \emptyset,\ \mathrm{i} = 0$ {Initialize}
While (halting criterion false) do

1. $\mathrm{i} = \mathrm{i} + 1$
2. $\mathbf{u} = \mathbf{\Phi}^*\mathbf{r}$ {Find the proxy}
3. $\Omega = \mathrm{supp}(\mathbf{u}_{2\mathrm{K}})$ {Identify index of largest 2K components}
4. $\Lambda = \Omega \cup \mathrm{supp}(\mathbf{x}_{\mathrm{i}-1})$ {Augment the index set}
5. $\mathbf{b}|_\Lambda = \mathbf{\Phi}_\Lambda^\dagger \mathbf{y}\ ;\quad \mathbf{b}|_{\Lambda^c} = \mathbf{0}$ {Signal estimate by least squares}
6. $\mathbf{x_i} = \mathbf{b_K}$ {Prune the signal estimation}
7. $\mathbf{r} = \mathbf{y} - \mathbf{\Phi x_i}$ {Update the new residual}

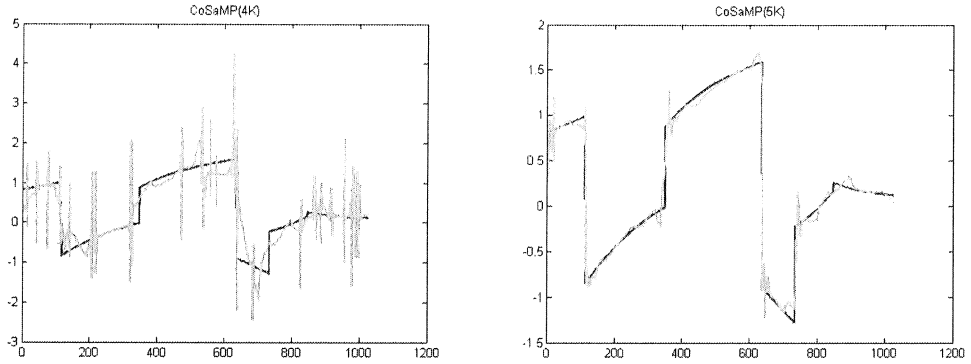
end while
return $\hat{\mathbf{x}} = \mathbf{x_i}$

3.2.2 Experiments

Same setup as for the OMP algorithm, we generate a signal of length N=1024 by using Matlab to implement. The recovery results with different size of M are displayed in the Figure 3.5. As compared with the OMP algorithm, the performance of CoSaMP is not as accurate as OMP algorithm. It is easy to tell when M=5K the reconstruction by OMP is almost exact while result of CoSaMP has some deviations by comparison to the original signal.



(a) (b)



(c) (d)

Figure 3.5: Example performance of CoSaMP signal recovery. The red dotted line is the orignal signal, the green solid line is the signal recovery. (a) Signal recovery from M=2K samples. (b) Signal recovery from M=3K samples. (c) Signal recovery from M=4K samples. (d) Signal recovery from M=5K samples.

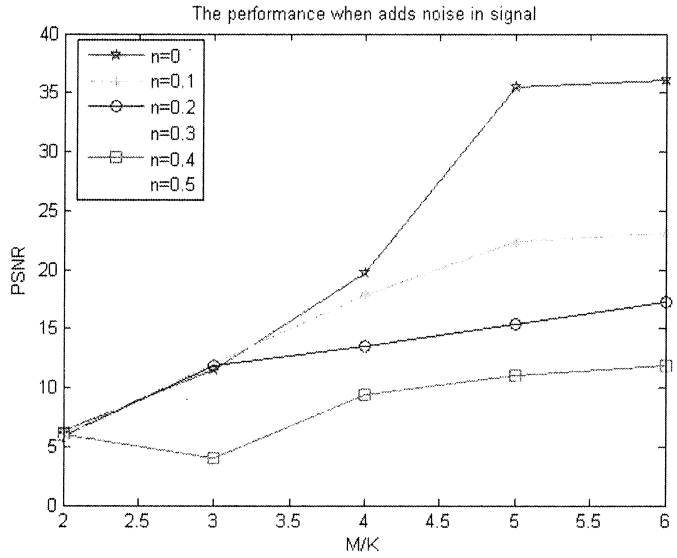


Figure 3.6: The stability tests of the CoSaMP algorithm when random noise is added to the original signal. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

Figure 3.6 and 3.7 show the performances of recovery of CoSaMP algorithm when adding random noise in signal and measurements respectively. If the original signal is perturbed by the noise and the noise power level is less than 0.1, the PSNR is more than 20 dB when M is bigger than 5K. If the noise power level is more than 0.1, the result is not good no matter how big M is. The performance of CoSaMP is better if we add noise to the measurements. The recovery results are acceptable if the noise power level is not more than 0.2 for M is bigger than 4K.

CoSaMP is more stable than OMP algorithm. If the signal or the measurements is interrupted by little noise, we can get an approximately accurate result with a large M.

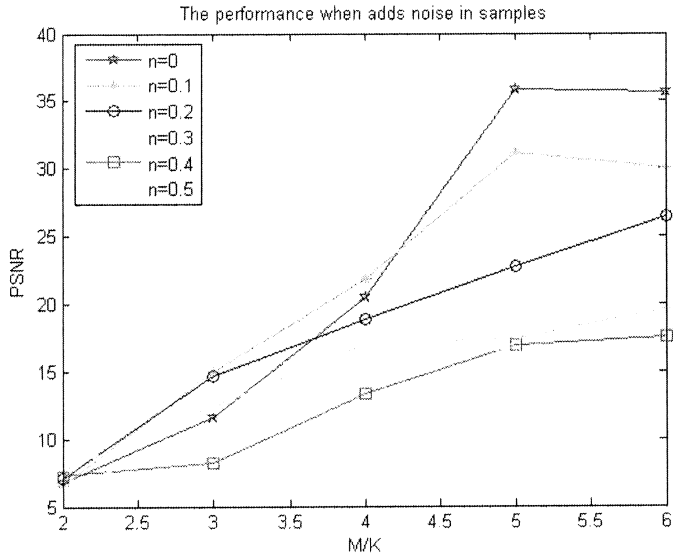


Figure 3.7: The stability tests of the CoSaMP algorithm when random noise is added to the measurements. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

Now let us compare the PSNR of OMP and CoSaMP and the results are given in Figure 3.8. Without the noise, the recovery results by OMP are much more accurate than the results by CoSaMP. For example, the PSNR of OMP is around 78 dB when M=5K while the PSNR of CoSaMP is about 32 dB. Figure 3.9 shows the comparison about runtimes between these two algorithms. CoSaMP recovery is extremely fast. When M=5K, CoSaMP needs less than 0.2 seconds to do the reconstruction but OMP requires more than 1.5 seconds. So if we only need an acceptable result and require more about the cost of calculation, CoSaMP algorithm is a better choice.

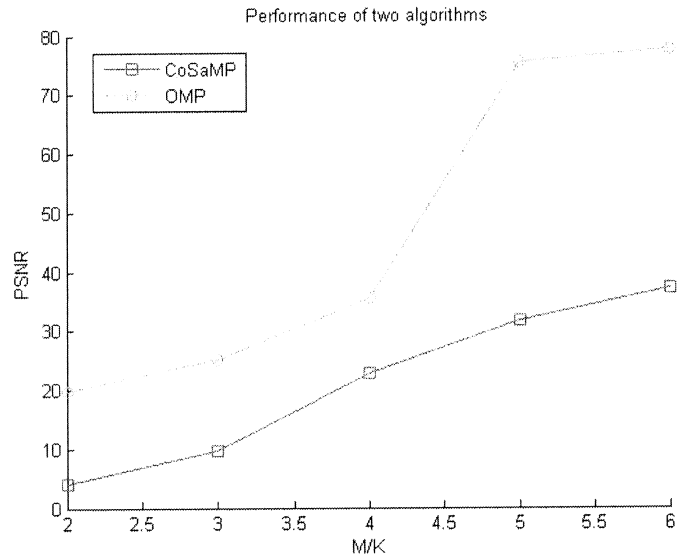


Figure 3.8: The comparison about performance of two algorithms

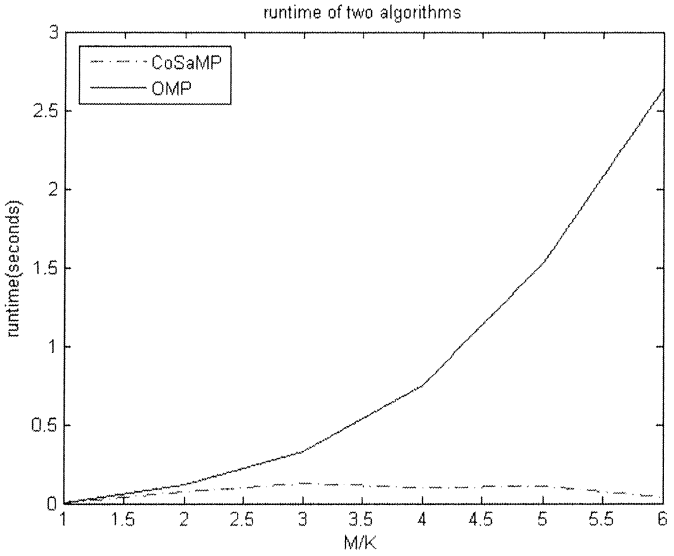


Figure 3.9: The comparison about runtime of two algorithms

3.2.3 Conclusion

Compared with OMP, CoSaMP algorithm can't achieve such a high PSNR as OMP when there is no noise. But CoSaMP has its own advantages. It is more stable when adding noise to the signal. The recovery result is acceptable if the noise is not too high and measurements are large. The calculation is extremely fast.

3.3 Model-based compressive sensing

3.3.1 Description

As we discuss before, in order to get the robust reconstruction, we need $\mathrm{M} = \mathcal{O}(\mathrm{Klog}(\mathrm{N}/\mathrm{K}))$ measurements. While there is a fact that modern wavelet image coders not only have most of small wavelets coefficients, but also the locations of the small part of large coefficients have a particular structure. One of the structured sparsity model have the large wavelet coefficients lived on a rooted, connected tree structure[17], the other one accounts for the large coefficients are clustered together[7,8]. Baraniuk and Duarte [21] based on the structural dependencies between the locations and values of signals and proposed a new signal acquisition method called model-based recovery. If the inter-dependency structure among the coefficients is considered in the process of reconstruction, fewer measurements will be required to offer the robust recovery. This is the foundation of model-based compressive sensing.

Structured sparse signals

A structured sparsity model $\mathcal{M}_\mathrm{K}$ [5] is defined as

$$\mathcal{M}_\mathrm{K} = \cup_{\mathrm{m}=1}^{\mathrm{m}_\mathrm{K}} \mathcal{X}_\mathrm{m} \ , \ \text{s.t.} \ \mathcal{X}_\mathrm{m} \coloneqq \{\mathbf{x} : \mathbf{x}|_{\Omega_\mathrm{m}} \in \mathbb{R}^\mathrm{K}, \mathbf{x}|_{\Omega_\mathrm{m}^\mathrm{C}} = 0\} \tag{7}$$

where $\mathbf{x}|_\Omega$ denotes the entries of $\mathbf{x}$ equaling to the set of indices $\Omega \subseteq \{1, \dots, \mathrm{N}\}$ and $\{\Omega_1, \dots, \Omega_{\mathrm{m}_\mathrm{K}}\}$ contains all the allowed support with $\#\Omega_\mathrm{i} = \mathrm{K}$. Ω^C represents the

complement of the set Ω.

$\mathcal{M}_K$ is the union all the K-dimensional subspaces and the signals from $\mathcal{M}_K$ are called K-structured sparse.

Model-based Restricted Isometry Property

For all $\mathbf{x} \in \mathcal{M}_K$, an $M \times N$ matrix $\mathbf{\Phi}$ has the $\mathcal{M}_K$-restricted isometry property if

$$(1 - \delta_{\mathcal{M}_K})\|\mathbf{x}\|_2^2 \leq \|\mathbf{\Phi x}\|_2^2 \leq (1 + \delta_{\mathcal{M}_K})\|\mathbf{x}\|_2^2 \quad (8)$$

Blumensath and Davies[10] quantified the number of measurements M required for reconstruction to have $\mathcal{M}_K$-restricted isometry property and pointed out that for any $t > 0$ and positive constant c, a random $M \times N$ i.i.d. subgaussian matrix $\mathbf{\Phi}$ has the $\mathcal{M}_K$-restricted isometry property with probability at least $1 - e^{-t}$ if

$$M \geq \frac{2}{c\delta_{\mathcal{M}_K}^2}\left(\ln(2m_K) + K \cdot \ln\frac{12}{\delta_{\mathcal{M}_K}} + t\right). \quad (9)$$

By substituting $m_K = \binom{N}{K} \approx (Ne/K)^K$, inequality (9) can be used for the bound of standard RIP.

As we know, m_K is much smaller than $\binom{N}{K}$ since m_K arises from the structure imposed. So the number of measurements needed that satisfied the $\mathcal{M}_K$-RIP is much fewer than that of the standard RIP.

Structured compressible signals

Richard [5] defined the algorithm to obtain the best K-term structured sparse approximation of $\mathbf{x}$ by

$$\mathbb{M}(\mathbf{x}, K) = \arg\min_{\hat{\mathbf{x}} \in \mathcal{M}_K} \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \quad (10)$$

If the ℓ_2 error about the best K-term structured sparse approximation $\hat{\mathbf{x}}$ and signal $\mathbf{x}$ is

$$\sigma_{\mathcal{M}_K}(\mathbf{x}) := \inf_{\hat{\mathbf{x}} \in \mathcal{M}_K} \|\mathbf{x} - \hat{\mathbf{x}}\|_2,$$

then

$$\sigma_{\mathcal{M}_K}(\mathbf{x}) = \|\mathbf{x} - \mathbb{M}(\mathbf{x}, K)\|_2$$

So the set of structured compressible signal is define as

$$\mathfrak{M}_s = \left\{x \in \mathbb{R}^N : \sigma_{\mathcal{M}_K}(\mathbf{x}) \leq GK^{-s}, 1 \leq K \leq N, G < \infty\right\}.$$

$x \in \mathfrak{M}_s$ is called an s-structured compressible signal under the structured sparsity model $\mathcal{M}_K$. The value of s is selected by minimizing the distance between $\sigma_{\mathcal{M}_K}(\mathbf{x})$ and GK^{-s}.

Enlarged union of subspaces

The enlarged union of subspaces is defined as

$$\mathcal{M}_K^B = \{\mathbf{x} = \sum_{r=1}^{B} \mathbf{x}^{(r)}, \text{with } \mathbf{x}^{(r)} \in \mathcal{M}_K\}$$

It is easy to tell that $\mathcal{M}_K = \mathcal{M}_K^1$ and $\mathcal{M}_K^B \subset \mathcal{M}_{BK}$.

$\mathbb{M}_B(\mathbf{x}, K)$ is the algorithm to obtain the best approximation of $\mathbf{x}$ in the enlarged union of subspaces and defined as

$$\mathbb{M}_B(\mathbf{x}, K) = \arg\min_{\hat{\mathbf{x}} \in \mathcal{M}_K^B} \|\mathbf{x} - \hat{\mathbf{x}}\|_2$$

Rooted tree structure

In a rooted tree, there is a unique path linking any two nodes. For a node k, all nodes that lie from k to the root are called ancestors of k; all nodes that lie from k away from the root are named descendants of k. The parent of k is the ancestor that links directly to k and the child

of k is the node that has k as its parent. A node can have several children but only one parent. Nodes without children are called the leaves. The parent of k is denoted as $p(k)$.

Wavelet coefficient tree

The subspace of functions in $L^2(\mathbb{R})$ of the form

$$f_i(t) = \sum_j s_{i,j}\varphi(2^i t - j)$$

is denoted V_i, where $\varphi(2^i t - j)$ is the scaling functions at scale 2^{-i}. The dilated, translated and normalized scaling functions are $\varphi_{i,j} = 2^{i/2}\varphi(2^i t - j)$. So every function $f_i \in V_i$ can be written as

$$f_i(t) = \sum_j s_{i,j}\varphi_{i,j}(t).$$

The general multiresolution analysis gives that a function ψ is said to be a wavelet if the detail space W_0 spanned by the functions $\psi(t-j)$ complements V_0 in V_1 [6]. Wavelets $\psi(t-j)$ should be a Riesz basis for W_0. So any $f_1 \in V_1$ can be decomposed as $f_1 = f_0 + d_0$, where $f_0 \in V_0$ is an approximation of f_1 and $d_0 \in W_0$ contains the lost details.

The detail spaces W_i are defined as the set of functions of the form

$$d_i(t) = \sum_j w_{i,j}\,\psi_{i,j}(t)$$

$\psi_{i,j}(t) = 2^{i/2}\psi(2^i j - t)$ is the dilated and translated wavelets.

According to the above, any function $f_{j+1} \in V_{j+1}$ can be decomposed as $f_{j+1} = f_j + d_j$, where $f_j \in V_j$ and $d_j \in W_j$.

Now consider a signal $\mathbf{x}$ at a finest scale I, we can write $\mathbf{x_I} \in V_I$ as

$$\mathbf{x_I} = d_{I-1} + d_{I-2} + \cdots + d_{i_0} + \mathbf{x_{i_0}}$$

$$= \sum_{i=i_0}^{I}\sum_j w_{i,j}\,\psi_{i,j} + \sum_j s_{i_0,j}\varphi_{i_0,j}$$

by repeating the decomposition $x_{i+1} = x_i + d_i$ until a certain level i_0.

Since $\mathbf{x_{i_0}} \to 0$ as $i_0 \to -\infty$ and $\mathbf{x_I} \to \mathbf{x}$ as $I \to \infty$, the wavelet representation of $\mathbf{x}$ is given by [6]

$$\mathbf{x} = \sum_{i,j} w_{i,j}\,\psi_{i,j}\,.$$

Using a matrix to illustrate, $\mathbf{x} = \boldsymbol{\psi}\mathbf{s}$ where $\boldsymbol{\psi}$ is a matrix containing the wavelet functions as the columns and $\mathbf{s} = \left[w_{0,0}, w_{1,0}, w_{1,1}, w_{2,0}, w_{2,1}, \ldots\right]^{\mathbf{T}}$ is a vector with the wavelet coefficients as the components. The vector $\mathbf{s}$ can form a rooted binary wavelet tree with $w_{0,0}$ as the root of the tree. Figure 3.10 shows the wavelet coefficients tree.

If a coefficient $w_{i,j} \in T$ and its parent $w_{i-1,\lfloor j/2 \rfloor} \in T$ as well, the coefficients satisfy this property will form a connected subtree. Then the definition of the structured sparsity model $\mathcal{T}_K$ is

$$\mathcal{T}_K = \{\mathbf{x} = \sum_{i,j} w_{i,j}\,\psi_{i,j} : w|\Omega^C = 0, |\Omega| = K\}$$

Here Ω is the index set. In order to recover the tree based signal, we need to solve the optimal problem as follows:

$$x_K^{\mathcal{T}} = \arg\min_{\bar{\mathbf{x}} \in \mathcal{T}_K} \|\mathbf{x} - \bar{\mathbf{x}}\|_2 \qquad (11)$$

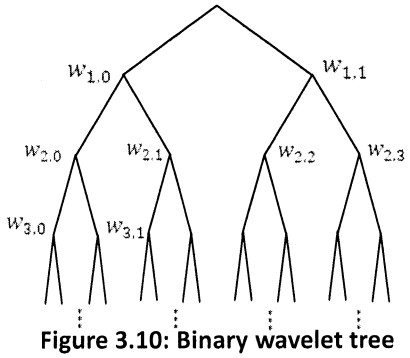


Figure 3.10: Binary wavelet tree

The structured sparsity model $\mathcal{T}_{\mathrm{K}}$ and optimal problem (10) are similar to the structured sparsity model $\mathcal{M}_{\mathrm{K}}$ and algorithm $\mathbb{M}(\mathbf{x}, \mathrm{K})$. Condensing sort and select is an algorithm that can solve the optimal problem (10) and (11). Because $\bar{\mathbf{x}}$ is K-sparse, we need to find K largest absolute value of wavelet coefficients $\mathrm{w_{i,j}}$ to get the optimal value of signal $\mathbf{x}$.

The CSSA algorithm

If we refer to $B(k)$ as the value of node k and $\psi(k)$ as the kernel of linear program, the condensing sort and select algorithm (shorted for CSSA) [16] is a greedy algorithm to find the maximum value of $B(\mathrm{k})\psi(k)$. The kernel should satisfy the nonincreasing constraint that is $\psi(k^*) \leq \psi(p(k^*))$.

(1) Use $\mathrm{k}^* = \mathrm{argmax}\{\mathrm{B(k)}: \psi(k) = 0\}$ to find the node k^* that has the largest data value of all of the nodes with $\psi(k) = 0$.

(2) If $\psi\big(p(k)\big) = 1$, $\psi(k^*)$ is set to be 1. While if $\psi\big(p(k)\big) = 0$, we merge k^* with $p(k^*)$ to form a larger supernode S by

$$\mathrm{v(S)} = \frac{\mathrm{B(k^*) + B(p(k^*))}}{2},$$

and $\mathrm{v(S)}$ is the supernode value (SNV) of the supernode S.

Supernode can also contain several nodes. Let $\mathrm{n(S)}$ denote the number of internal nodes the supernode S has and $\psi(\mathrm{S})$ denote the kernel value, its SNV is

$$\mathrm{v(S)} = \frac{1}{\mathrm{n(S)}} \sum_{\mathrm{k} \in \mathrm{S}} \mathrm{B(k)}$$

(3) If $\psi(p(s))$ is still equal to zero, we continue to merge nodes into supernodes until we get

$$\mathrm{S}^* = \arg\max \{\mathrm{v(S)}: \psi(\mathrm{S}) = 0\}$$

with $\psi\big(\mathrm{p(S^*)}\big) = 1$, then the kernel of S^* can be set to 1.

Before given the algorithm of CSSA, the explanation of the notations is:

$\mathrm{B(k)}$: The value of node k.

ψ: The kernel value.

γ: The volume of kernel.

Γ: The volume counter.

$\mathrm{p(0)}$: Root of the tree.

The algorithm of CSSA is as follows:

Algorithm 3: CSSA

Input: value of nodes $B(k)$, tree, $\gamma \geq 0$
Output: ψ

$v(k) = B(k), n(k) = 1, \psi(k) = 0,$
$\Gamma = 0, \psi(p(0)) = 1$ {Initialize}
while $\Gamma \leq \gamma$ do
1. $S^* = \arg\max \{v(S): \psi(S) = 0\}$ {identify the largest supernode}
2. If $\psi(p(S^*)) = 1$
   $\psi(S^*) = \min(1, (\gamma - \Gamma)/n(S^*)); \ \Gamma = \Gamma + n(S^*)$ {update the kernel value}
3. else $\psi(p(S^*)) = 0$
   Merge S^* and $p(S^*)$ into a single supernode {condense}
   end if

end while

Model-based signal reconstruction algorithm

As we have already known, the model-based signal recovery algorithm is based on the CoSaMP algorithm, so we can get the model-based algorithm by replacing the best K-term sparse approximation step in CoSaMP with a best K-term structured sparse approximation. Because each time, we merely search m_K subspaces of $\mathcal{M}_K$ that is significantly fewer than $\binom{N}{K}$ subspaces of $\sum_K$ in the conventional recovery algorithms. Fewer measurements are needed to keep the robust of algorithm.

Baraniuk and Duarte gave the proof of performance of structured sparse signal recovery. By given a noise measurement $\mathbf{y} = \mathbf{\Phi x} + \mathbf{e}$, the signal estimation $\hat{\mathbf{x}}_i$ of structured sparse signal $\mathbf{x} \in \mathcal{M}_K$ obtained from iteration i satisfies

$$\|\mathbf{x} - \hat{\mathbf{x}}_i\|_2 \leq 2^{-i}\|\mathbf{x}\|_2 + 15\|\mathbf{e}\|_2$$

As the growing of the number of iteration, the ℓ_2-error is becoming smaller and smaller. In the absence of noise, the model-based algorithm can recover a structured sparse signal with high accuracy.

The Model-based algorithm has 5 major steps during each iteration:

(1) Find the proxy of the current samples' residual.
(2) Obtain the best approximation of the proxy in the enlarge union of $\mathcal{M}_K^2$ and unite the index of the best approximation with the index of the current signal approximation.
(3) Estimate the signal on the merged set of components by using the least squares.
(4) Choose the best K-term structured sparse approximation of the estimation as the signal approximation to prune the signal estimation
(5) Update the samples' residual.

Algorithm 4: Model-based

Input: Measurement matrix $\mathbf{\Phi}$, measurement $\mathbf{y}$, sparsity level K of the ideal signal $\mathbf{x}$, structured sparse approximation algorithm $\mathbb{M}$

Output: K-sparse estimate $\hat{\mathbf{x}}$, index set Λ, measurement estimate $\mathbf{a_m}$, residual $\mathbf{r} = \mathbf{y} - \mathbf{a_m}$

$\mathbf{x}_0 = 0,\ \mathbf{r_0} = \mathbf{y},\ \Lambda_0 = \emptyset,\ \mathrm{i} = 0$ {Initialize}

While (halting criterion false) do

1. $\mathrm{i} = \mathrm{i} + 1$
2. $\mathbf{u} = \mathbf{\Phi}^*\mathbf{r}$ {Find the proxy}
3. $\Omega = \mathrm{supp}(\mathbb{M}_2(\mathbf{u}, \mathrm{K}))$ {Prune residual estimate based on structure}
4. $\Lambda = \Omega \cup \mathrm{supp}(\mathbf{x}_{\mathrm{i}-1})$ {Augment the index set}
5. $\mathbf{b}|_\Lambda = \mathbf{\Phi}_\Lambda^\dagger \mathbf{y}\ ;\ \ \mathbf{b}|_{\Lambda^c} = \mathbf{0}$ {Signal estimate by least squares}
6. $\mathbf{x_i} = \mathbb{M}(\mathbf{b}, \mathrm{K})$ {Prune the signal estimation}
7. $\mathbf{r} = \mathbf{y} - \mathbf{\Phi}\mathbf{x_i}$ {Update the new residual}

end while

return $\hat{\mathbf{x}} = \mathbf{x_i}$

3.3.2 Experiments

By using the same method as OMP to generate a test signal with length N=1024. Figure 3.11 and 3.12 give the stability tests of Model-based algorithm with different power levels of noise. Compared with OMP and CoSaMP algorithms, Model-based recovery is much more robust. Even with noise, the results of recovery are ~~till permissible,~~ satisfactory especially when the noise interrupt the measurements. That means it is possible to get an approximately accurate reconstructed signal with an interference sample.

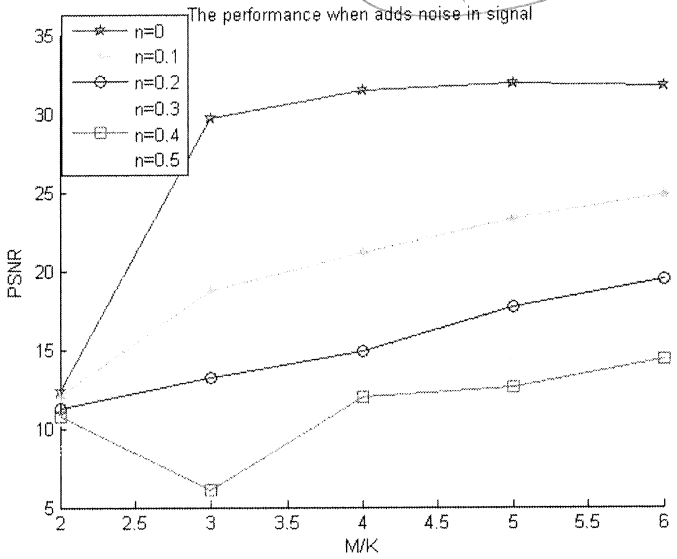


Figure 3.11: The stability tests of the Model-based algorithm when we add random noise to the original signal. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

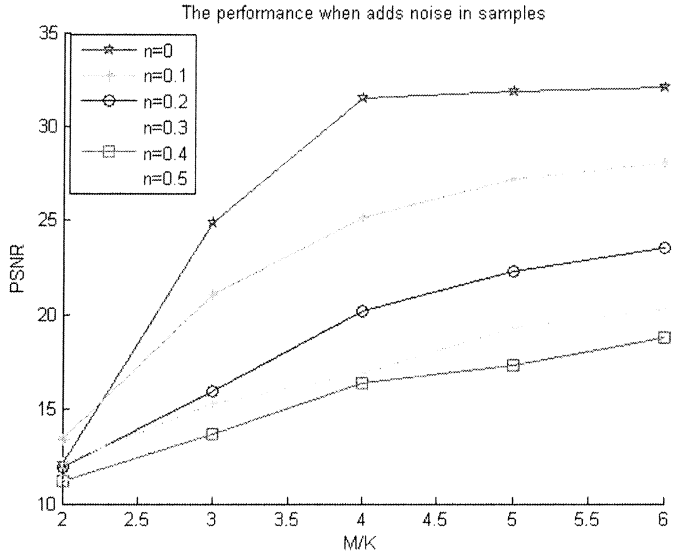
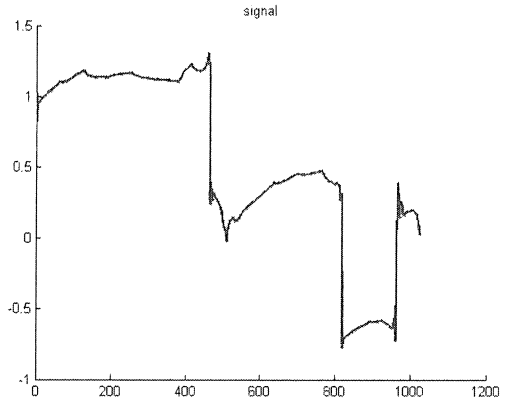


Figure 3.12: The stability tests of the Model-based algorithm when random noise is added to the measurements. The noise is a random generated vector with the same size as the signal. The noise power level is increased from 0 to 0.5 step by step.

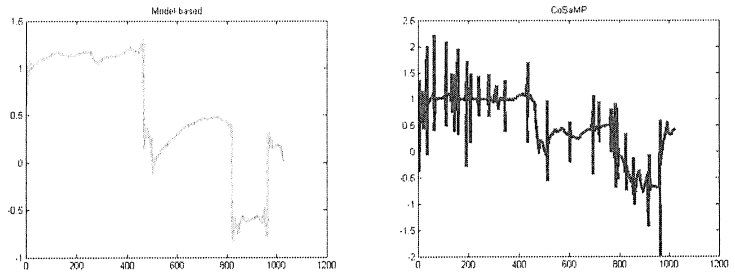
Figure 3.13 depicts the experiment about the performance of CoSaMP and Model-based algorithms without noise. It is easy to see that M=2K is far fewer than the requirement of CoSaMP to recover the signal accurately. On the contrary, model-based recovery using M=2K can get a pretty good result.

Figure 3.14 illustrates how the number of measurements M affects the PSNR of two algorithms. Model-based recovery achieves good recovery at M=3K while CoSaMP gets this performance at M=5K. It is not difficult to tell that comparing with CoSaMP, Model-based recovery uses significantly fewer measurements to offer the same stability.

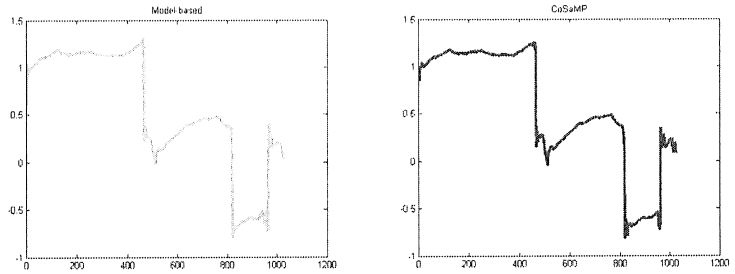
By using the same class of signals, the recovery time of Model-based and CoSaMP algorithms are compared empirically and the results are illustrated in Figure 3.15. In general, Model-based recovery is slower than CoSaMP. The best benefits of model-based recovery are obtained at M=4K. Around this area, Model-based algorithm yields much higher PSNR than CoSaMP and the computational times of two methods are comparable.



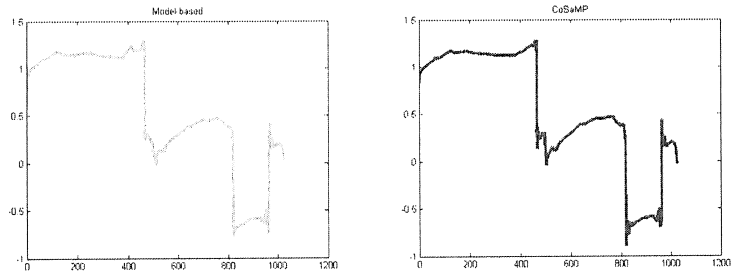
(a) The original signal of length N=1024



(b) Signal recovery from M=2K=126 samples



(c) Signal recovery from M=4K=252 samples



(d) Signal recovery from M=6K=378 samples

Figure 3.13：The comparison about CoSaMP and Model-based algorithms. (a) is the original signal with length N=1024. The signal is K-sparse in wavelet basis and K=63. (b), (c) and (d) are the comparisons of reconstruction with different size of M. Left column is the result of Model-based algorithm and right column is the result of CoSaMP algorithm.

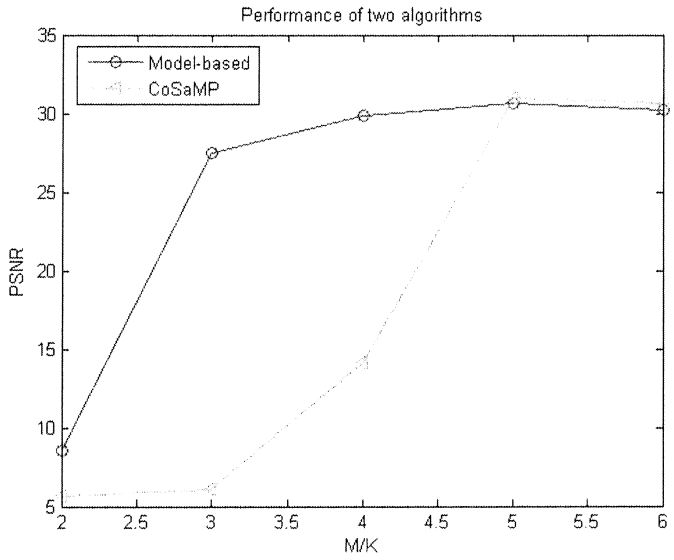


Figure 3.14: The comparison about performance of Model-based and CoSaMP algorithms

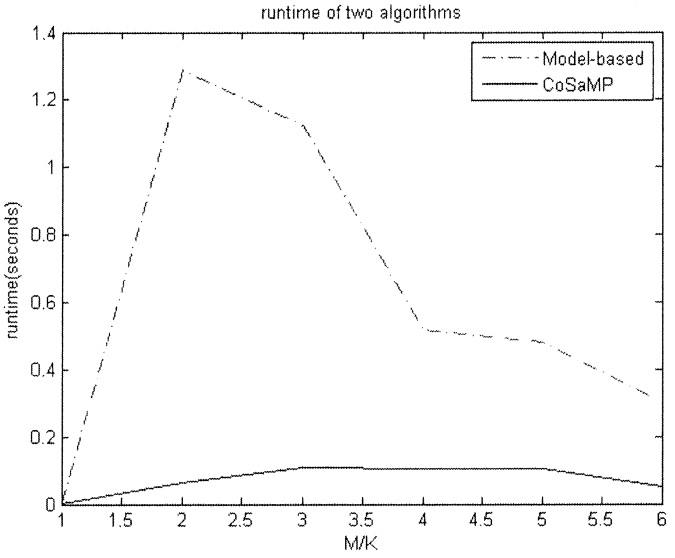


Figure 3.15: The comparison about runtime of Model-based and CoSaMP algorithms

Now, Figure 3.16 and Figure 3.17 give the comparisons of stability and recovery time of OMP, CoSaMP and Model-based algorithms. With the noise free measurements, OMP can achieve a really high PSNR when M is larger than 5K. But as we have discussed in section 3.1.2, OMP recovery is very unstable. If the signal or measurements are disturbed by noise, the recovery is not accurate any more even with a large M. When M is less than 4K, Model-based algorithm can get the highest PSNR comparing with the other algorithms. In general, CoSaMP is the fastest algorithm. This is due to the fact that CoSaMP uses the simple K-term approximation. This algorithm picks the largest K components each time instead one component in OMP algorithm and the CSSA step in Model-based recovery is more computational. So OMP is the slowest algorithm and Model-based is slower.

Comparing Figure 3.16 and 3.17, the near perfect point Model-based algorithm obtains is around M=4K. The recovery error of Model-based approach at this point is smaller than the other algorithms and ~~runtime~~ is acceptable.

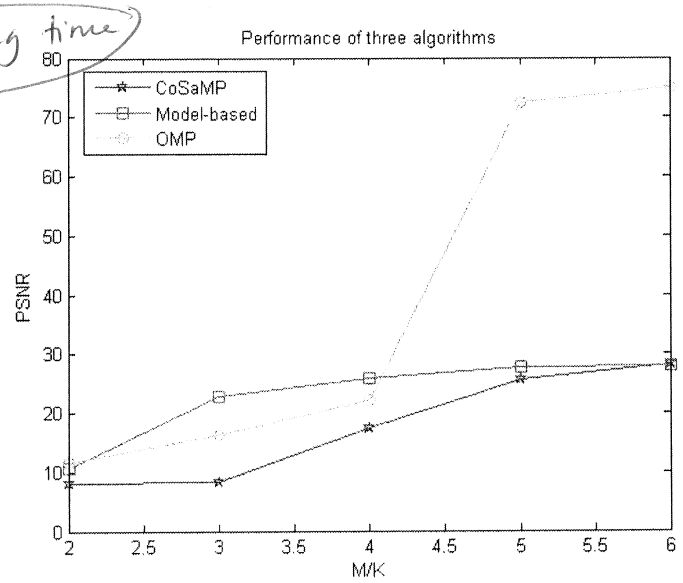


Figure 3.16: The comparison about performance of three algorithms

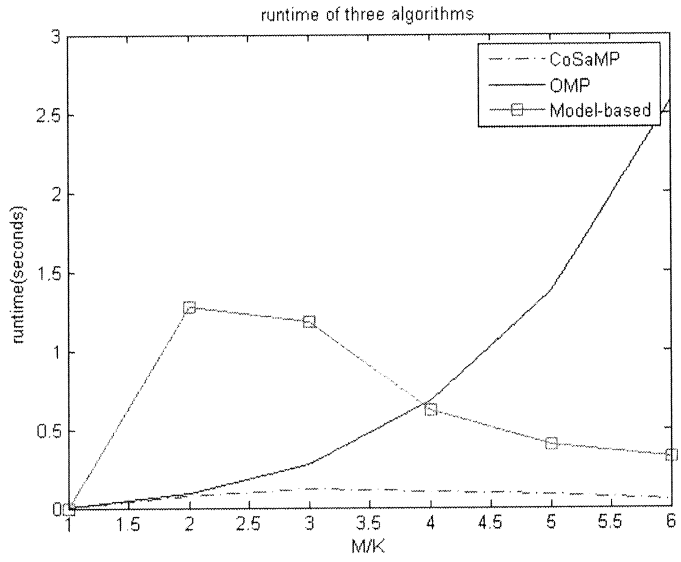


Figure 3.17: The comparison about runtime of three algorithms

3.3.3 Conclusion

From the above discussions, Model-based algorithm can recover a signal accurately with fewer measurements. This approach is more stable. When the samples are interfered by noise, it is possible to reconstruct the signal properly. The recovery time is much faster than OMP but a litter slower than CoSaMP. Overall, Model-based is a costless, effective and efficient ~~algorithm~~.

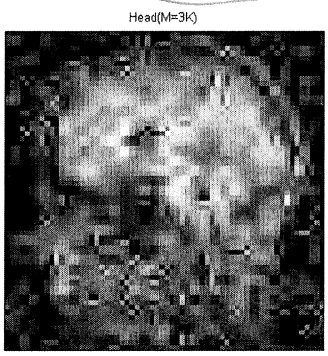
4 Experiments on MRI Images

In this chapter, the algorithms mentioned above will be tested by series real 2D medical images. These images are provided by MRI scanner. MRI stands for "magnetic resonance imaging" and is also known as spin imaging. MRI makes use of the property of Nuclear magnetic resonance (NMR) to image nuclei of atoms inside the body and this technique is widely used in medical imaging to visualize detailed internal structures. The main problem is ~~that MRI~~ that MRI scanner requires a huge number of measurements to reconstruct an image, so it will take a long time of scanning. Comparing to X-ray and CT (computed tomography), MRI does not expose the patient to the hazards of ionizing radiation. But the long time scanning is still harmful and will bring a lot of ~~uncomfortable~~ to the patient. Compressed sensing can significantly reduce the number of samples, so it is broadly used in medical image processing.

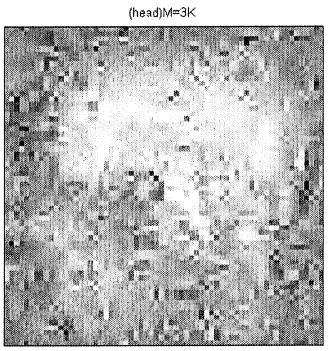
Since OMP is very unstable, it is not accurate if the signal is perturbed slightly by noise. This algorithm is also slow so it is quite expensive compared with the other algorithms. It is not practical in reality. So in this part, we only test CoSaMP and Model-based algorithms and compare ~~the~~ performance ~~with~~ different types of images. All of the test images in this thesis are of size $\mathrm{N} = 256 \times 256$. But ~~according~~ to the data processing capability of my computer, I resize the image to be one quarter of the original one and the new images are of size $\mathrm{N} = 64 \times 64$. Because the size of images is small, the resolution of test images is ~~not quite high~~ and the quality of reconstructed results is ~~not quite~~ good ~~as well~~. But it is not difficult to tell the difference when we change the sample size M and compare the performance by using different methods.

Head

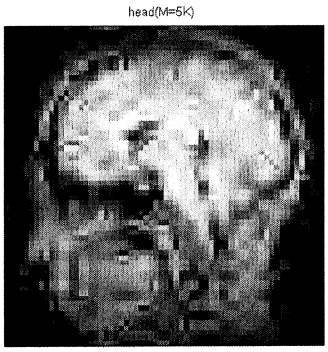
First, an example of head image recovery is given in Figure 3.18. There are a lot of details in this type of image. In this figure, both PSNR are not higher than 20 dB until $\mathrm{M} \geq 5\mathrm{K}$. The recovery qualities for both algorithms are ~~not quite~~ good.

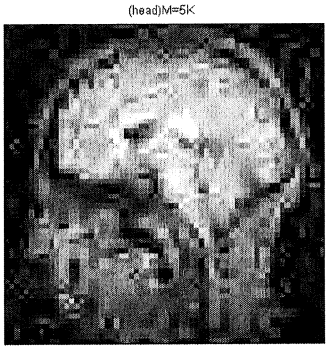


M=3K, PSNR=17.31dB

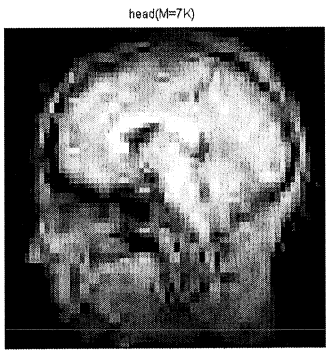


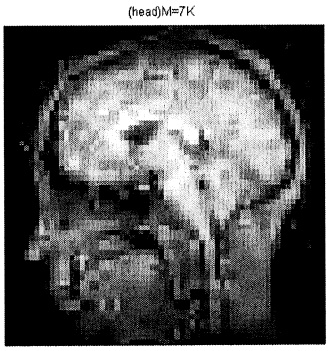
M=3K, PSNR=13.84dB





M=5K, PSNR=21.15dB **M=5K, PSNR=20.17dB**





M=7K, PSNR=22.8dB **M=7K, PSNR=22.88dB**

Figure 3.18: Example of Model-based and CoSaMP recovery on images. The left column is Model-based recovery and the right column is CoSaMP recovery.

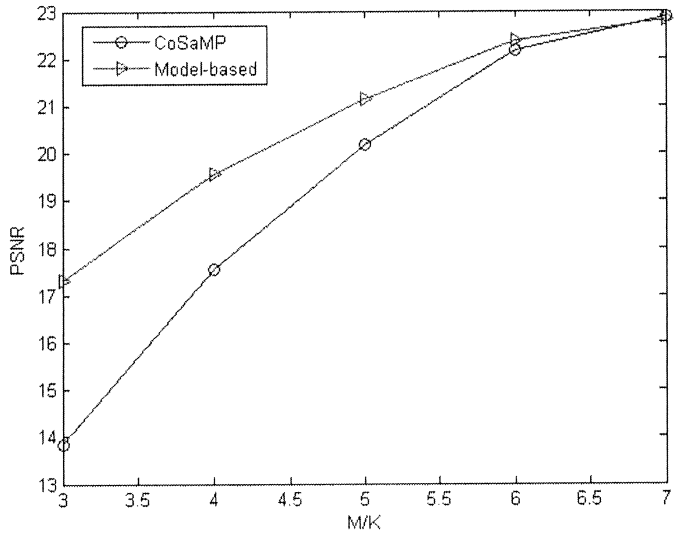
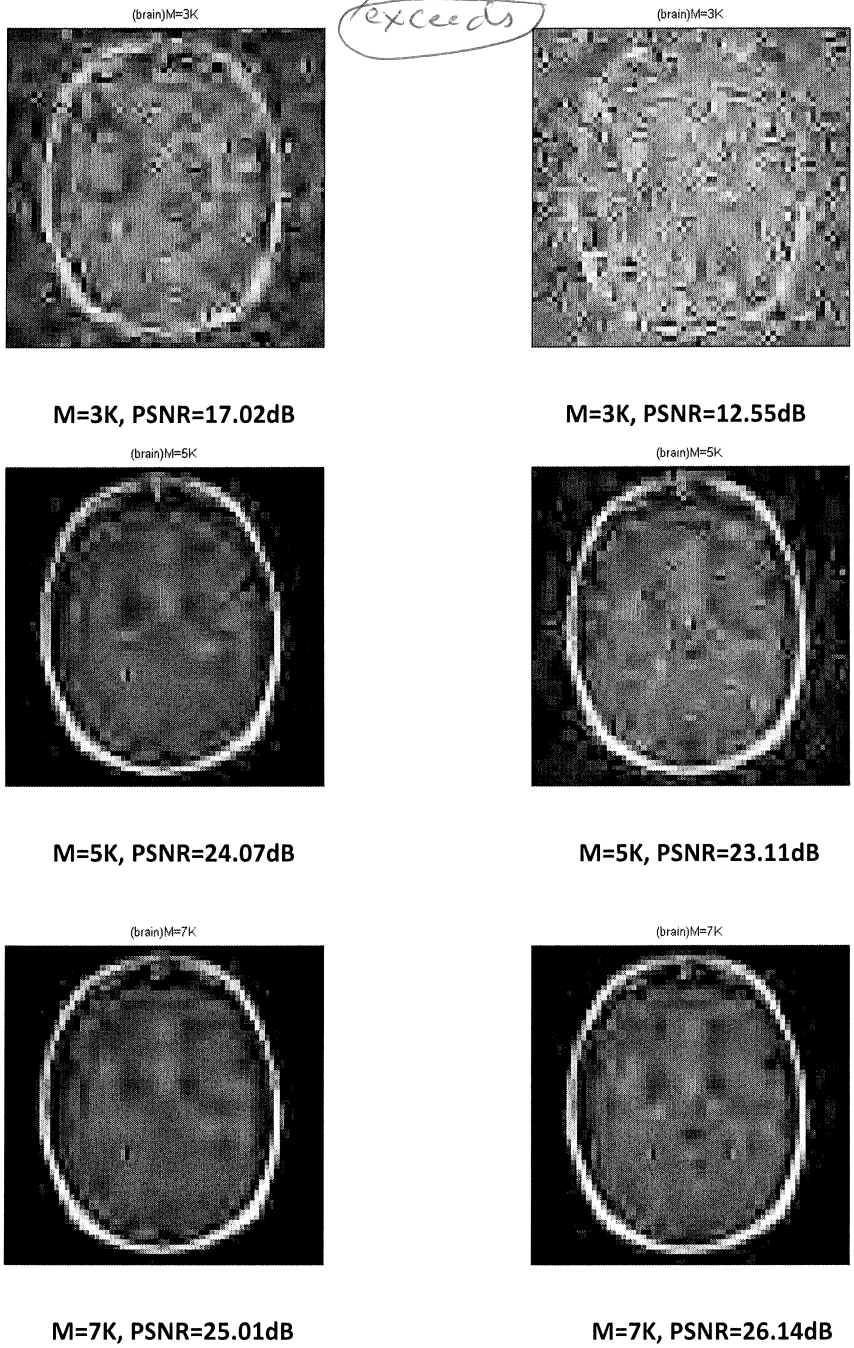


Figure 3.19: Performance of Model-based and CoSaMP recovery on a 2D head image.

Brain I

The following is the test results of brain (horizontal view of a head which is cut into two pieces). This type of image is quite smooth in the centre, but it has a bright circle which is quite different

from the other parts. Model-based recovery can achieve a very good PSNR when $M \geq 4K$ while the PSNR of CoSaMP recovery is not acceptable until $M \geq 5K$. But the PSNR of CoSaMP is increasing very fast as the growing of M. From the curve in Figure 3.21, the PSNR of CoSaMP is higher than Model-based recovery when M is above $5.5K$.



M=3K, PSNR=17.02dB **M=3K, PSNR=12.55dB**

M=5K, PSNR=24.07dB **M=5K, PSNR=23.11dB**

M=7K, PSNR=25.01dB **M=7K, PSNR=26.14dB**

Figure 3.20: Example of Model-based and CoSaMP recovery on images. The left column is Model-based recovery and the right column is CoSaMP recovery.

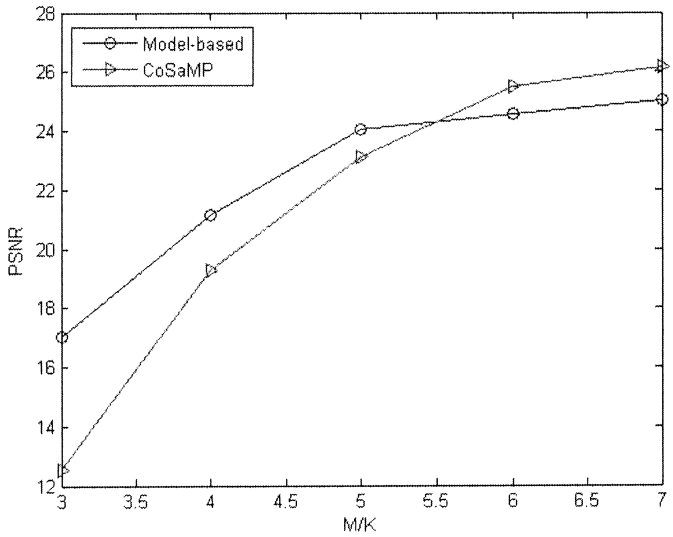
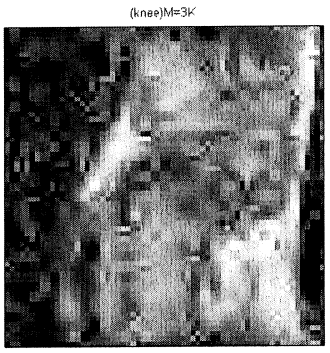
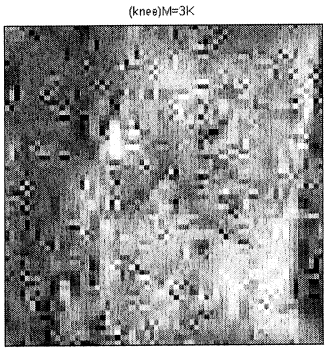


Figure 3.21: Performance of Model-based and CoSaMP recovery on a 2D brain image.

Knee

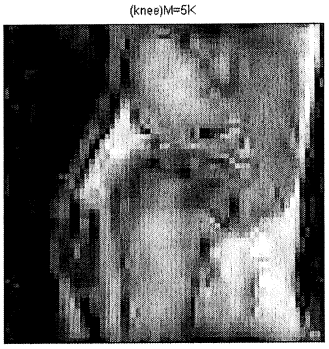
On the whole, the knee image is quite smooth and doesn't have a lot of curves. The main trend of PSNR is similar as the first type of brain. Both PSNR are growing fast as M increases and the PSNR of CoSaMP is bigger than Model-based at the end when M is larger than 6K. The qualities of recovery are good when $M \geq 4K$.

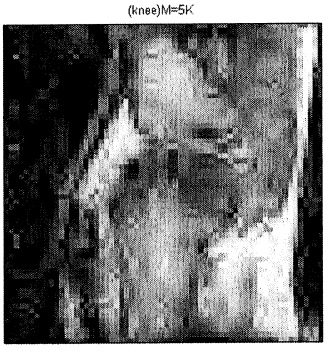




M=3K, PSNR=19.33dB

M=3K, PSNR=15.33dB





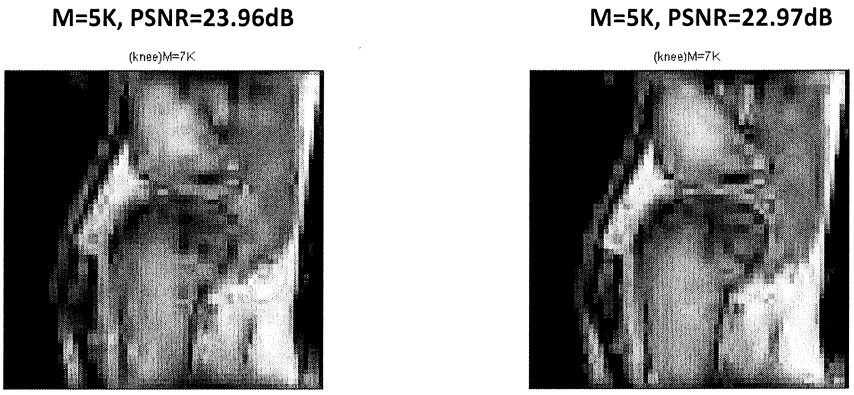


Figure 3.22: Example of Model-based and CoSaMP recovery on images. The left column is Model-based recovery and the right column is CoSaMP recovery.

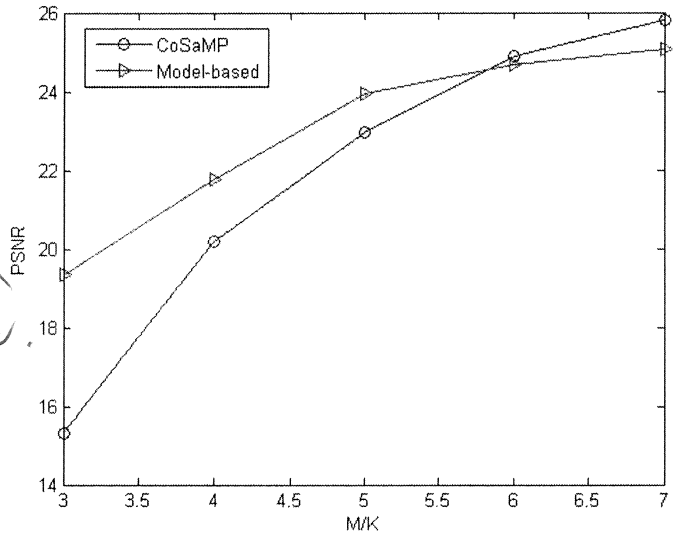
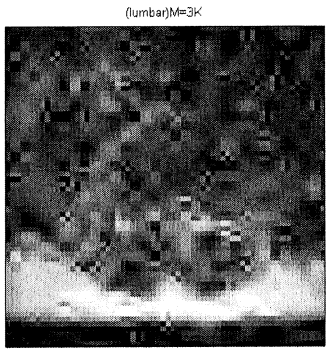


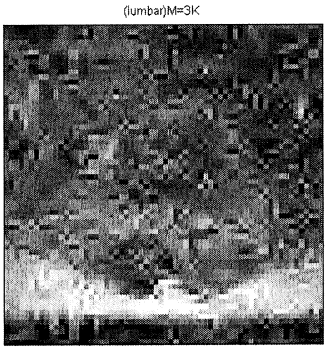
Figure 3.23: Performance of Model-based and CoSaMP recovery on a 2D knee image.

Lumbar

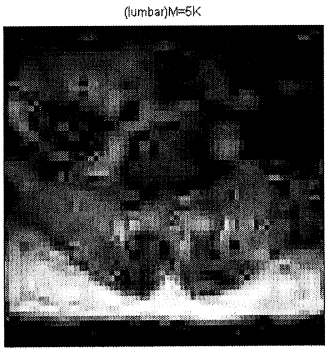
There are some clear objects in the lumbar images and these objects are very smooth. According to the Figure 3.24 and 3.25, Model-based recovery achieves quite good results and the PSNR of this algorithm is more than 20dB when $M = 3K$. The PSNR of CoSaMP recovery is very low at $M = 3K$, and it can get a comparable PSNR when $M \geq 5K$. Finally, the quality of CoSaMP algorithm exceeds Model-based at $M = 7K$.



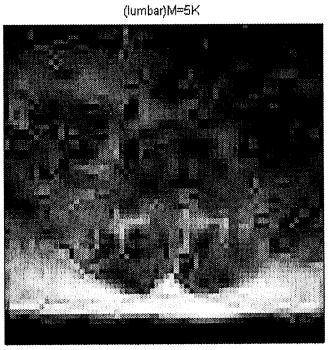
M=3K, PSNR=20.39dB



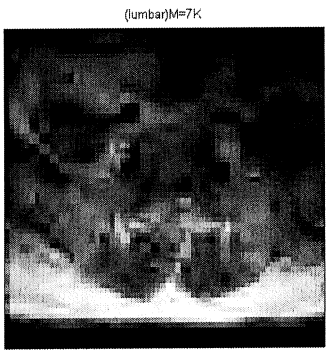
M=3K, PSNR=16.70dB



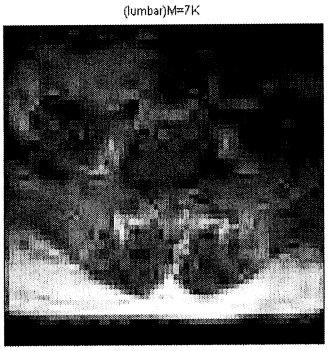
M=5K, PSNR=24.56dB



M=5K, PSNR=23.93dB



M=7K, PSNR=26.33dB



M=7K, PSNR=26.63dB

Figure 3.24: Example of Model-based and CoSaMP recovery on images. The left column is Model-based recovery and the right column is CoSaMP recovery.

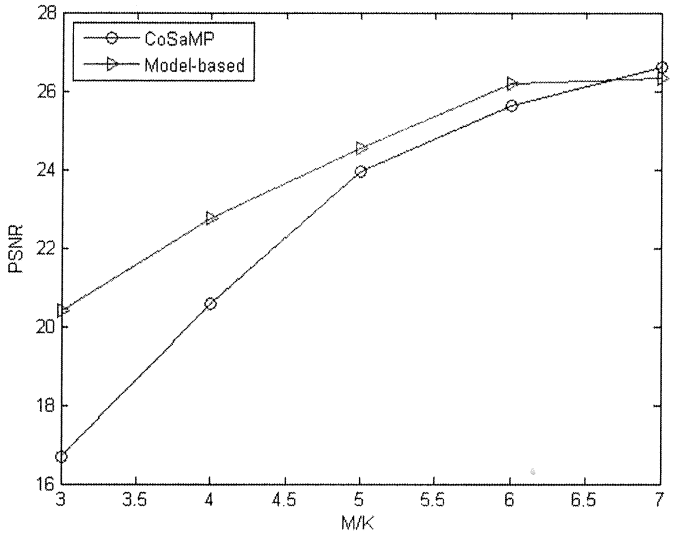
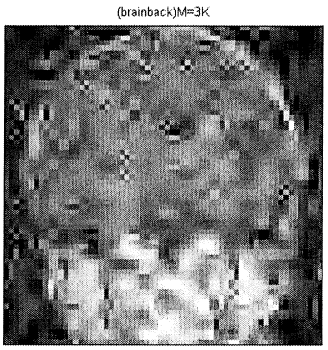


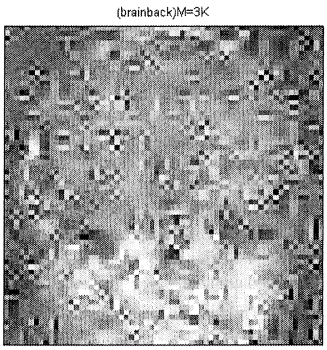
Figure 3.25: Performance of Model-based and CoSaMP recovery on a 2D lumbar image.

Brain II

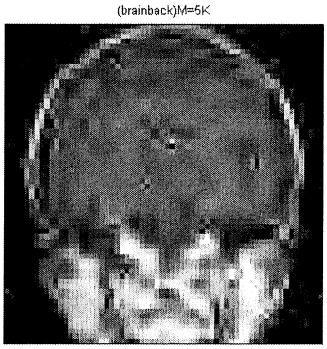
This is the second type of brain image and it is the vertical view of a head which is cut into two pieces. The top of this image is quite smooth but the bottom part is rough. The results of the experiments are illustrated in Figure 3.26 and 3.27. The PSNR of both algorithms are not more than 20dB until $M \geq 5K$.



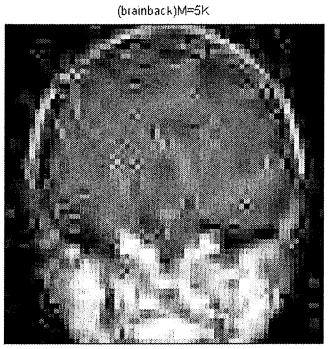
M=3K, PSNR=17.08dB



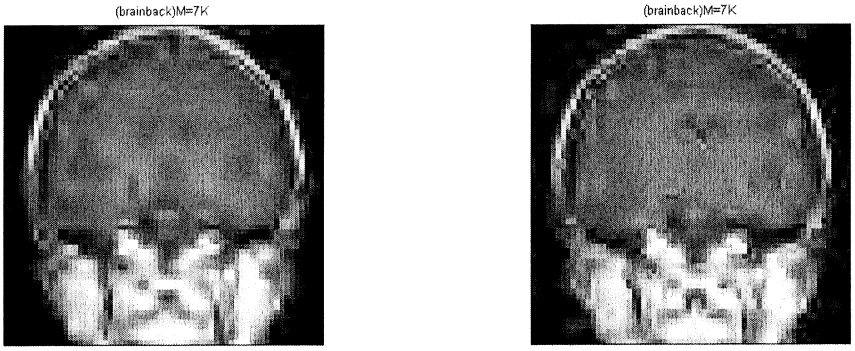
M=3K, PSNR=12.87dB



M=5K, PSNR=21.61dB



M=5K, PSNR=21.09dB



M=7K, PSNR=23.18dB **M=7K, PSNR=23.72dB**

Figure 3.26: Example of Model-based and CoSaMP recovery on images. The left column is Model-based recovery and the right column is CoSaMP recovery.

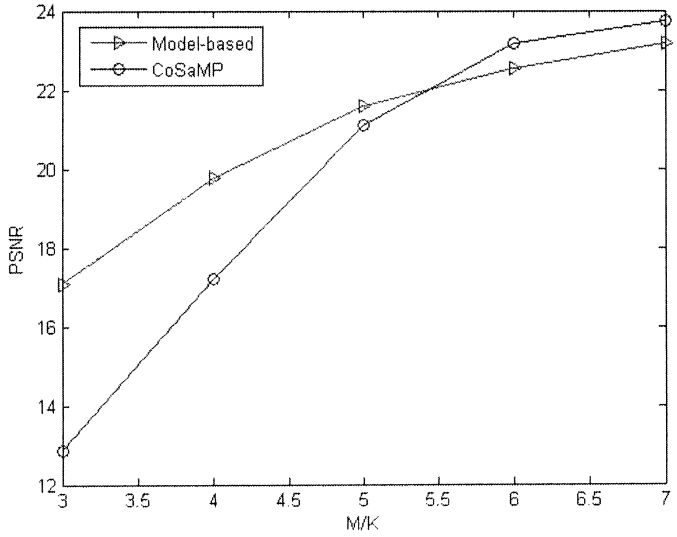


Figure 3.27: Performance of Model-based and CoSaMP recovery on a 2D brain image.

5 Conclusion

The traditional process of image compression is quite costly. It acquires the entire signal at beginning, then does the compression and throws most of the information away at the end. The new idea of image compression combines signal acquisition and compression as one step which improves the overall cost significantly.

The process of new image compression is to find a measurement matrix $\mathbf{\Phi} \in \mathbb{R}^{M \times N} (M < N)$ and multiply with the signal $\mathbf{x} \in \mathbb{R}^{N}$ that we want to compress in order to get M linear measurements $\mathbf{y} = \mathbf{\Phi x}$. $\mathbf{y}$ is the compressed sample that we wish to get. Since $M < N$, the system is underdetermined if we want to reverse the process and reconstruct the signal $\mathbf{x}$. In theory, there are infinitely $\mathbf{x}$ that satisfy this system, so it seems impossible to reconstruct the signal. Fortunately, most of signals in reality are spare or spare under some basis (e.g. wavelets). If we can find the location of those non-zero entries, we can reconstruct the signal uniquely.

In my thesis, we discussed three reconstruction algorithms and compared the advantages and disadvantages of them. Orthogonal matching pursuit (OMP) is lacked in stability guarantees. Even it can achieve very high PSNR when the size of measurements M is large in the noise free case, it is not accurate any more if the signal or measurements are perturbed by noise. Since the algorithm picks the optimal entries one by one, it is very slow. So OMP is not an ideal algorithm in reality. Compressive Sampling Matching Pursuit (CoSaMP) is fastest among these three algorithms. The property of stability is better than OMP. When noise is added to the signal, the PSNR is acceptable if we have a large size of measurements M. Model-based algorithm is the most stable algorithm. Though the experiments of 1D and 2D, it is not difficult to tell model-based algorithm can offer a robust recovery by using fewer measurements comparing to CoSaMP. By testing the MRI images, both algorithms can provide satisfied results when the images are smooth. But when the images are rough or have a lot of details, the recovery results are not good. This kind of images needs more measurements to reconstruct the images.

Table 1: The comparison of three algorithms

OMP	☹ Not optimally fast. ☹ Lacked in uniform guarantees. ☹ Not stable.
	☺ PSNR is very high without noise.
CoSaMP	☹ Need a lot of measurements.
	☺ Fastest. ☺ Uniform guarantees. ☺ Possible to be stable.
Model-based	☺ Fast. ☺ Uniform guarantees. ☺ Large possibility to be stable. ☺ Need fewer measurements.

Reference

[1] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Trans. Inform. Theory, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[2] D. Donoho, "Compressed sensing," IEEE Trans. Inform. Theory, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[3] "Nyquist-shannon sampling theorem." Wikipedia, 2011.
http://en.wikipedia.org/wiki/Nyquist-Shannon_sampling_theorem.

[4] Justin Romberg, Michael Wakin, "Compressed Sensing: A Tutorial." IEEE Statistical Signal Processing Workshop, Aug. 26, 2007.

[5] Richard G. Baraniuk, Marco F. Duarte, "Model-Based Compressive Sensing." arXiv: 0808.3572v5 [cs. IT], Dec. 9, 2009.

[6] Joran Bergh, Fredrik Ekstedt, Martin Lindberg, "Wavelets." January 1999.

[7] M. Stojnic, F. Parvaresh, and B. Hassibi, "On the reconstruction of block-sparse signals with an optimal number of measurements," IEEE Trans. Signal Processing, vol. 57, no. 8, pp. 3075–3085, Aug. 2009.

[8] Y. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," IEEE Trans. Info. Theory, vol. 55, no. 11, pp. 5302–5316, Nov. 2009.

[9] J. Tropp and A.C. Gilbert, "Signal recovery from partial information via orthogonal matching pursuit," IEEE Trans. Inform. Theory, vol. 53, no. 12, pp. 4655-4666, 2007.

[10] T. Blumensath and M. E. Davies, "Sampling theorems for signals from the union of finite-dimensional linear subspaces," IEEE Trans. Info. Theory, vol. 55, no. 4, pp. 1872–1882, Apr. 2009.

[11] "Peak signal-to noise ratio." Wikipedia, 2011.
http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio

[12] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. A method for large-scale ℓ_1-regularized least-squares problems with applications in signal processing and statistics. Submitted for publication, 2007.

[13] 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. Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing, 1(4):586{598, 2007.

[14] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Comm. Pure Appl. Math., 57:1413-1457, 2004.

[15] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Info. Theory, 53(12):4655–4666, 2007.

[16] R. G. Baraniuk and D. L. Jones, "A signal-dependent time-frequency representation: Fast algorithm for optimal kernel design," IEEE Trans. Signal Processing, vol. 42, no. 1, pp. 134–146, Jan. 1994.

[17] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using hidden Markov models," IEEE Trans. Signal Processing, vol. 46, no. 4, pp. 886–902, Apr. 1998.

[18] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by Basis Pursuit. SIAM J.

Sci. Comput., 20(1):33–61, 1999.
[19] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. Sparse solution of underdetermined linear equations by stagewise Orthogonal Matching Pursuit (StOMP). Submitted for publication, 2007.
[20] S. Mallat and Z. Zhang. "Matching Pursuits with time-frequency dictionaries". IEEE Trans. Signal Process., 41(12):3397–3415, 1993.
[21] Richard G. Baraniuk and Marco F. Duarte. "Model-based Compressive Sensing". Information theory, IEEE Transactions, April 2010
[27] D. Needell and R. Vershynin. "Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit". Submitted for publication, October 2007.
[28] D. Needell and R. Vershynin. "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit". Submitted for publication, July 2007.
[29] D. Needell and J.A. Tropp. "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples". Applied and Computational Harmonic Analysis, 2009.
[30] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process., 41(12):3397–3415, 1993.
[31] B. K. Natarajan. Sparse approximate solutions to linear systems. SIAM J. Comput., 24:227–234, 1995.