วิธีค้นหาแบบไบนารีเพื่อใช้กำจัดสัญญาณรบกวนอิมพัลส์ในการสร้างกลับสัญญาณของคอมเพรสเซนซิง

นางสาวสุวิชญา สุวรรณวิมลกุล

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555

A BINARY SEARCH ALGORITHM FOR IMPULSIVE NOISE REMOVAL IN COMPRESSED SENSING RECONSTRUCTION

Miss Suwichaya Suwanwimolkul

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2012

Thesis Title	A BINARY SEARCH ALGORITHM FOR IMPULSIVE NOISE REMOVAL IN COMPRESSED SENSING RECONSTRUCTION.
By	Miss Suwichaya Suwanwimolkul
Field of Study	Electrical Engineering
Thesis Advisor	Assistant Professor Supatana Auethavekiat, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

…….. Dean of the Faculty of Engineering
(Associate Professor Boonsom Lerdhirunwong, Dr.Ing)

THESIS COMMITTEE

…… Chairman
(Assistant Professor Charnchai Pluempitiwiriyawej, Ph.D.)

…… Thesis Advisor
(Assistant Professor Supatana Auethavekiat, Ph.D.)

……. External Examiner
(Duangrat Gansawat, Ph.D.)

สุวิชญา สุวรรณวิมลกุล : วิธีค้นหาแบบไบนารีเพื่อใช้กำจัดสัญญาณรบกวนอิมพัลส์ในการสร้างกลับสัญญาณของคอมเพรสเซนซิง. (A Binary Search Algorithm for Impulsive Noise Removal in Compressed Sensing Reconstruction) อ. ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ.ดร. สุพัฒนา เอื้อทวีเกียรติ, 96 หน้า.

สัญญาณรบกวนอิมพัลส์ในสัญญาณบีบอัด (**y**) ส่งผลให้สัญญาณสปาร์สที่สร้างกลับมีการกระจายตัวพลังงานแตกต่างจากเดิม ในวิธี Approximated Measurement Preprocessing (AMP) ข้อมูลที่ใหญ่ที่สุดใน **y** จะถูกดึงออกไปจนกระทั่งการกระจายตัวพลังงานเป็นตามรูปแบบของภาพ วิทยานิพนธ์นี้เสนอวิธีกำจัดสัญญาณรบกวนแบบรวดเร็วสองวิธีคือ วิธีค้นหาขอบอย่างรวดเร็ว (GBF) และวิธีค้นหาความชันสูงอย่างรวดเร็ว (GSSF) เพื่อลดเวลาประมวลผลของวิธี AMP สัญญาณภาพถูกพิจารณาเป็นสัญญาณสปาร์สในโดเมน Discrete wavelet แบบ Octave อัตราส่วนระหว่างพลังงานนอกซับแบนระดับที่สาม และพลังงานทั้งหมด (*ER*) ถูกใช้เพื่อตรวจจับสัญญาณรบกวนอิมพัลส์ สัญญาณภาพมีความซ้ำซ้อนจึงสามารถสร้างกลับภาพที่มีคุณสมบัติใกล้เคียงกันแม้จะตัดสัญญาณที่ใหญ่ที่สุดบางตัวไปก็ตาม การค้นหาแบบไบนารีถูกนำมาใช้เพื่อหาจำนวนสัญญาณที่ถูกรบกวนให้อยู่ภายใน $+g$ ของจำนวนที่ถูกต้อง โดย g คือค่าคงตัวที่กำหนดไว้แล้ว ซึ่งเมื่อกำหนดให้ g เป็นร้อยละของขนาดของ **y** แล้วจำนวนครั้งของการสร้างกลับจะเป็นค่าคงที่ การค้นหาแบบไบนารีใน GBF และ GSSF ใช้เกณฑ์ค่า *ER* และอัตราการเปลี่ยนแปลง *ER* ตามลำดับ จากการเปรียบเทียบ GBF และ GSSF กับ AMP, การสร้างกลับด้วยฟังก์ชัน Huber (HUBER) และ Lorentzian Iterative Hard Thresholding (LIHT) ในการทดลองกับบล็อคภาพขนาด 16×16 พิกเซล จำนวน 100 บล็อค และภาพขนาด 256×256 พิกเซล จำนวน 20 ภาพ พบว่า GBF และ GSSF ให้ PSNR และคุณภาพภาพใกล้เคียงกับ AMP แต่ใช้เวลาการคำนวณน้อยกว่าเมื่อความน่าจะเป็นของการเกิดสัญญาณรบกวนมากกว่า 0.05 และ GBF, GSSF และ AMP ให้ผลดีกว่า HUBER และ LIHT GBF ให้ PSNR สูงกว่าและใช้เวลาการคำนวณน้อยกว่า GSSF ขณะที่ GSSF ให้ผลลัพธ์ที่ดีกว่าเมื่อสัญญาณรบกวนมีขนาดเล็ก ไม่ควรใช้ GBF และ GSSF ในกรณี (1) ไม่สามารถสร้างภาพเป็นสัญญาณสปาร์สด้วยวิธี Wavelet Shrinkage Thresholding หรือ (2) สัญญาณรบกวนมีขนาดเล็กกว่าข้อมูลที่ใหญ่ที่สุดใน **y** การผสานระหว่างอัตราส่วนพลังงานกับ HUBER จะถูกศึกษาต่อไปเพื่อขจัดสัญญาณรบกวนขนาดเล็ก

ภาควิชา วิศวกรรมไฟฟ้า ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมไฟฟ้า ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์ .....

ปีการศึกษา 2555

5470492321 : MAJOR ELECTRICAL ENGINEERING
KEYWORDS :COMPRESSED SENSING (CS)/ ROBUST ALGORITHM/ PREPROCESSING ALGORITHM/ BINARY SEARCH/

SUWICHAYA SUWANWIMOLKUL: A BINARY SEARCH ALGORITHM FOR IMPULSIVE NOISE REMOVAL IN COMPRESSED SENSING RECONSTRUCTION. ADVISOR: ASST. PROF. SUPATANA AUETHA-VEKIAT, Ph.D., 96 pp.

Impulsive noise in compressed measurement signal (**y**) leads to the reconstruction of the sparse signal whose energy distribution is different than the original signal. In Approximated Measurement Preprocessing (AMP), the highest elements in **y** are successively removed until the energy distribution conforms to the one of images. This thesis proposes two greedy algorithms, namely Greedy Boundary Finder (GBF) and Greedy Steep Slope Finder (GSSF), with an aim to reduce the computational cost of AMP. Images are assumed to be sparse in octave discrete wavelet domain. The ratio of energy outside L_3 subband and the total energy is used to detect the impulsive noise. Information in an image is highly redundant; therefore, some largest elements can be removed without causing severe degradation to the reconstruction result. Binary search is used to estimate the number of the noisy element to within $+g$ of the actual number, where g is the predefined constant. The number of the reconstruction is a fixed number, when g is set as the unit of the percent to the length of **y**. GBF and GSSF uses the energy ratio and the change of energy ratio as the cost function for binary search, respectively. GBF and GSSF were compared with AMP, the reconstruction with Huber penalty function (HUBER) and Lorentzian Iterative Hard Thresholding (LIHT). The experiment on 100 16×16 image blocks and 20 256×256 images revealed that GBF and GSSF provided the comparable PSNR and visual quality to AMP, and required less computational time when the noise probability was higher than 0.05. Furthermore, GBF, GSSF and AMP were better than HUBER and LIHT. GBF provided higher PSNR with lower computational cost than GSSF. However, GSSF was more robust when the noise magnitude was smaller than the largest element in **y**. GBF and GSSF were not efficient in case that (1) an image could not be sparsified by wavelet shrinkage thresholding or (2) the noise magnitude was smaller than the largest element in **y**. The integration of the energy ratio to HUBER is being investigated for the rejection of small noise.

Department : Electrical Engineering Student's Signature .....

Field of Study : Electrical Engineering Advisor's Signature .....

Academic Year : 2012

ACKNOWLEDGEMENTS

This work contained in this thesis represents the accumulation of one and a half years of work made possible only by the collective support of family, mentors, friends, colleagues. First, I would like to thank to my advisor, Asst. Prof. Dr. Supatana Auethavekiat, for her devotion and her valuable suggestion. I also would like to thank Dr. Perichat Sermwuthisarn for her invaluable suggestion and supports. It was an honor to study under their guidance and to share a moment of my life.

I also would like to thank Dr. Jitkomut Songsiri. The work on Huber cost function would not be possible without her guidance.

My thanks also to my thesis defense committees: Assst. Prof. Dr. Charnchai Pluempitiwiriyawej and Dr. Duangrat Gansawat. Their reviews and invaluable comments help make this thesis more complete.

Several other people helped in the production of this dissertation. I am grateful to Dr. Sopon Phumeechanya for help compiling the experimental data. I would like to thank Ms. Thanissorn Panarungsun and Ms. Hathaichanok Thummachirdchupong for their invaluable helps. Many thanks also to all colleagues at Multi Signal Processing Research Laboratory, Chulalongkorn University, for their academic help, technical/theory information and programming/data information.

Last but not least, my life has been fulfilled with love and support of my family. I am extremely grateful to my parents, Prabprueng Yuareekul and Witaya Suwanwimunkun, for their love, caring, understanding and preparing me for my future life since I was a kid. This work is dedicated to them.

.

CONTENTS

Page

Page

List of Tables

List of Figures

Figure Page

CHAPTER I

INTRODUCTION

1.1 Introduction

Compressed Sensing (CS) is a paradigm for the sampling and reconstruction of a compressible signal at the rate lower than the Nyquist rate [1-3]. A signal is defined to be compressible if it can be represented by a few nonzero elements in some domain. The domain is not definitely defined; therefore, most signals can be considered compressible. CS has a wide range of applications including hyperspectral imaging [4] , super-resolution [5] and medical image processing [6-7]. The reconstruction from a compressed signal in CS is an underdetermined problem. The sparsity of an original signal and an incoherency between the sparse and the sensing domains are the major keys to achieve the unique reconstruction [1]. The reconstruction is set up as a problem to find the sparsest signal that can construct the input compressed signal. The sparsity, which is the number of non-zero elements, can be measured by L_0 norm; however, the minimization of L_0 norm is a non-deterministic polynomial-time (NP) hard problem, so it is either heuristically approximated by L_1 norm [8-9], or replaced by heuristic rules [10-14].

In the reconstruction from a noisy signal, it is often assumed that the noise is bound and the CS reconstruction becomes a constraint minimization problem [8-9]. The noise tolerance can be increased by (1) choosing the penalty function to conform to the noise in the system and (2) adding the model of a signal into the reconstruction. The most popular penalty function is L_2 norm, but L_∞ norm [15] or a function in the robust statistic [16-19] can also be used to increase the tolerance of the bounded noise.

Though most noise is bounded, it is not guaranteed that all noise will be bounded such as the impulsive noise during the thunder storm or power surge. In case of the unbounded noise, the error becomes very large and the constraint minimization

of the popular L_2-norm penalty function leads to incorrect reconstruction. To reduce the effect of the large error, a function from robust statistics is often used as the penalty function [16-19]. It is found that the model of generalized Cauchy distribution (GCD) in an M-estimator is tolerant to impulsive noise [16]. The Lorentzian function is often used in robust CS reconstruction [16-18], because it is the special case of GCD. However, Lorentzian function is not convex; therefore, the global minimum is not guaranteed. The greedy algorithm for the reconstruction using Lorentzian function is also proposed [18]. It is based on an iterative hard thresholding (IHT) [14]. However, IHT requires a high measurement rate; hence, it is suitable only for a very sparse signal. The Huber function, which is another popular function in robust statistics, is also used in CS reconstruction [19]. The optimization of Huber function can be formulated as a convex optimization problem and can be solved by many existing algorithms. However, the value of Huber function is not bound; thus, its performance is degraded when the noise is large or heavily populated.

Signal models can be used to reduce the effect of noise. The model of an image signal is used to detect the existence of impulsive noise in [20]. An image signal has most of its energy contained inside a low frequency domain. The failed reconstruction due to the impulsive noise leads to large energy leaking out of the low frequency domain [20]. The ratio of leakage energy and the total energy of the reconstructed sparse signal is used to detect the existence of impulsive noise. The method in [20], defined as Approximated Measurement Preprocessing (AMP), rejects the impulsive noise by successively removing the largest elements until the ratio is within the predefined threshold. In AMP, the elements corrupted by impulsive noise are considered to have much larger magnitude than the noise-free elements, because the magnitude of impulsive noise is very large. The major benefits of AMP over the reconstruction by robust statistic are the simpler implementation and the tolerance to large noise. However, the computational time is not low, because the number of the CS reconstruction applied in AMP is the same as the number of noise levels.

1.2 Objectives

1. To reduce the computational time of AMP, while the simplicity and the large noise tolerance are preserved.
2. To investigate the limitation of detecting the impulsive noise by the ratio between the leakage energy and the total energy.

1.3 Problem Statements

The image signal is highly redundant, it is unnecessary to remove the exact number of noisy elements. Hence the redundancy in an image can be exploited to reduce the computational time, since some noiseless elements can be removed without affecting the visual quality of the reconstruction.

1.4 Contributions

1. This thesis proposes the integration of a binary search and AMP to reduce the computational time. The redundancy in image signal is exploited. The binary search is used to detect the number of noisy elements to within the value of g of the actual number, where g is the predefined constant and has the unit of percent of the size of a compressed measurement signal.
2. This thesis investigates the limitation of the impulsive noise detection by measuring the ratio of the leakage energy and the total energy.

1.5 Scopes

1. The proposed algorithm is for gray-scale image only.
2. The algorithm is applicable to images that can be sparsified by octave-tree DWT with db8 as a mother wavelet.
3. The compressed measurement signal is corrupted by impulsive noise only.

4. The magnitude of impulsive noise must be at least three times higher than the peak of a signal, y_{max}. When the distribution of the noise magnitude is Gaussian with the standard deviation (SD) of the y_{max}, the lowest magnitude mean is $5y_{max}$. The value of $5y_{max}$ is used to ensure that most noise will be higher than $3y_{max}$ (within 2 SD) and there is a little chance that the noise will be smaller than the signal. The assumption that the noise must be larger than the signal is based on that the impulsive noise is very large.

1.6 Research Procedures

1. Study previous researches relevant to the dissertation.

   Researches on CS

   Researches on impulsive noise removal in CS

2. Design and implement the proposed algorithm.

3. Evaluate the proposed algorithm by the following datasets.

   100 image blocks of 16×16 pixels. The result is evaluated by %*MSE* and computational time.

   10 standard test images of 256×256 pixels. The result is evaluated by PSNR, computational time and visual quality.

4. Check whether the conclusions meet all the objectives of the dissertation.

5. Write the dissertation.

CHAPTER II

BACKGROUND

This chapter addresses the related research of the dissertation. It is divided into three sections. The Compressed Sensing (CS) theory is described in Section 2.1. Since the CS reconstruction method used in this thesis is Orthogonal Matching Pursuit with Partially Known Support (OMP-PKS), it is described in detail in Section 2.2. The CS reconstruction in impulsive noise environment is then described in Section 2.3.

2.1 Compressed Sensing (CS)

CS is a compression technique at a sub-Nyquist rate [1-3]. It is applicable to a compressible signal which is defined as a signal that is sparse in some domain. It is based on a sparse representation and an incoherence property. The sparse representation implies compressibility, while the incoherence property implies the isometric mapping in reconstruction process.

Most natural signals are compressible, because they can be transformed by an orthonormal basis expansion to become sparse. The sparsity is defined as the number of non-zero elements in the sparse representation, i.e. a signal is *k*-sparse, when its sparse representation has *k* non-zero elements.

CS consists of two processes: (1) compression and (2) reconstruction processes. In the compression process, the sparse signal is compressed using a measurement matrix. The compressed signal is called the compressed measurement signal in CS. The compression of a compressible signal, $\mathbf{x} \in \mathbf{R}^N$, is summarized as the following linear equation.

$$\mathbf{y} = \mathbf{\Phi s} = \mathbf{\Phi \Psi}^T \mathbf{x}, \tag{2.1}$$

where $\mathbf{y} \in \mathbf{R}^M$ is a compressed measurement signal ($M << N$);

$\mathbf{\Phi} \in \mathbf{R}^{M \times N}$ is a random measurement matrix;

$\mathbf{\Psi} \in \mathbf{R}^{N \times N}$ is a predefined matrix whose column corresponds to the basis of a sparse domain;

$\mathbf{s} \in \mathbf{R}^{N}$ is a sparse signal.

Without loss of generality, $\mathbf{x}$ is assumed to be sparse in this proposal. Consequently, $\mathbf{\Psi}$ is an identity matrix, and $\mathbf{x}$ and $\mathbf{s}$ can be used interchangeably.

In the reconstruction process, $\mathbf{x}$ is estimated from $\mathbf{y}$. Since the dimension of $\mathbf{x}$ is larger than the one of $\mathbf{y}$, the reconstruction is an underdetermined problem. Two additional criteria are introduced to guarantee the high probability of the successful reconstruction.

1) Restricted Isometric Properties (RIP)

RIP is introduced to guarantee the almost orthogonal property of $\mathbf{\Phi}$. It is defined as the following equation.

$$(1-\delta_k)\|\mathbf{x}\|_2^2 \le \|\mathbf{\Phi x}\|_2^2 \le (1+\delta_k)\|\mathbf{x}\|_2^2 \tag{2.2}$$

$\delta_k > 0$ is the k-restricted isometry constant and $\|\cdot\|_2$ is the L_2 norm.

2) Measurement bound

The measurement bound provides the smallest $\mathbf{y}$ that has a high probability of successful reconstruction. The size of the smallest $\mathbf{y}$ is defined according to the following equation.

$$M = ck\log N, \tag{2.3}$$

where $c \ge 1$ is a constant;

k is the number of non-zero elements in the sparse signal.

a priori is often introduced to solve an underdetermined problem. In CS, *a priori* is that the signal is sparse. The sparsity can be measured by L_0 norm; thus, the reconstruction of CS can be written as the following problem.

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathbf{R}^N} \|\mathbf{x}\|_0 + \alpha f(\mathbf{\Phi x} - \hat{\mathbf{y}}) \tag{2.4}$$

where $\|\cdot\|_0$ is the L_0 norm;

α is a regularization parameter;

$f(\cdot)$ is a penalty function.

There are various choices of $f(\cdot)$ depending on how the residual error is penalized. However, the L_2 norm is often used.

Since the minimization of L_0 norm is an NP-hard problem, L_1 norm is used as the heuristic estimation of sparsity [8, 9]. CS reconstruction becomes the optimization of the following problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathbf{R}^N} \|\mathbf{x}\|_1 + \alpha f(\mathbf{\Phi x} - \hat{\mathbf{y}}) \tag{2.5}$$

where $\|\cdot\|_1$ is the L_1 norm.

If $f(\cdot)$ is convex, Equation (2.5) can be solved by many existing efficient convex optimization algorithms.

In a greedy CS reconstruction algorithm, the minimization of L_0 norm is replaced by heuristic rules [10-14]. One of popular rules is that a non-zero element has a high correlation between its corresponding basis in $\mathbf{\Phi}$ and $\mathbf{y}$ [10-11]. After the index of the non-zero element has been found, the value of the non-zero element is estimated such that $f(\mathbf{\Phi x} - \hat{\mathbf{y}})$ is the lowest.

There are many ways to improve a noise tolerance in CS reconstruction. For example, a penalty function from robust statistics is used as $f(\cdot)$ to increase the noise tolerance [16-19]; a signal model can be included into the penalty function as $f(\mathbf{\Phi x} - \hat{\mathbf{y}} + \mathbf{Ex})$, where $\mathbf{Ex}$ is the function measuring how $\mathbf{x}$ compiles with the predefined model [21]; the model of a signal can be incorporated directly into the reconstruction algorithm [10-14].

2.2 OMP-PKS [12]

OMP-PKS is adapted from OMP [11], which is one of the most popular greedy algorithms for CS reconstruction. OMP-PKS has the same benefits as OMP in that it has low computational time and its implementation is simple; however, it has higher stability and requires a lower measurement rate (M/N) than OMP.

When the original signal is sparsified by a wavelet transform, LL subband contains most of the important information and should be selected as the known support in OMP-PKS. The algorithm of OMP-PKS for the reconstruction of a sparse signal in the wavelet domain can be summarized as follows.

Input

- The number of the non-zero elements (the sparsity level), k
- A random measurement matrix, $\mathbf{\Phi} \in \mathbf{R}^{M \times N}$
- A compressed measurement signal, $\mathbf{y}$

The selection of known supports

1) Define a set containing the indices of elements (bases) in LL subband as $\Gamma = \{\gamma_1, \gamma_2, \cdots, \gamma_{|\Gamma|}\}$, where γ_i is the index of the i^{th} element in LL subband, and $|\Gamma|$ is the size of Γ. The process to select every element in LL subband as non-zero elements is as follows.

- Set the number of the selected non-zero elements to $|\Gamma|$: $t = |\Gamma|$.
- Set an index set of non-zero elements to Λ: $\Lambda_t = \Gamma$.
- Set a matrix containing the bases of non-zero elements as follows:

$$\mathbf{\Phi}_t = \begin{bmatrix} \phi_{\gamma_1} & \phi_{\gamma_2} & \cdots & \phi_{\gamma_{|\Gamma|}} \end{bmatrix},$$

where ϕ_i is the i^{th} column in $\mathbf{\Phi}$.

2) Estimate the values of the non-zero elements by the following least squared problem.

$$\mathbf{z}_t = \arg\min_{\mathbf{z}} \left\| \mathbf{y} - \mathbf{\Phi}_t \mathbf{z} \right\|_2$$

3) Calculate the compressed measurement signal estimated by $\mathbf{z}_t$ as $\mathbf{a}_t$. Then find the residual, $\mathbf{r}_t$.

$$\mathbf{a}_t = \mathbf{\Phi}_t \mathbf{z}_t$$
$$\mathbf{r}_t = \mathbf{y} - \mathbf{a}_t$$

The selection of the remaining non-zero elements

1) Terminate the algorithm if the number of the selected non-zero elements, t, is not less than k.

2) Find the index of the non-zero element as the index of the column in $\mathbf{\Phi}$ that has the highest correlation to $\mathbf{r}_t$.

$$\lambda_{t+1} = \arg\max_{j=1,\ldots,N} \left| \left\langle \mathbf{r}_t, \phi_j \right\rangle \right|$$

If there are multiple columns with the highest correlation, select one deterministically.

3) Update t and construct Λ_t and Φ_t.

- The number of the selected element is increased by 1: $t = t + 1$.
- The set containing the indices of the non-zero elements is constructed as follows:

$$\Lambda_t = \left[\Lambda_{t-1} \quad \lambda_t \right].$$

- The matrix containing the bases of nonzero elements is constructed as follows:

$$\mathbf{\Phi}_t = \left[\mathbf{\Phi}_{t-1} \quad \phi_{\gamma_t} \right].$$

4) Solve the following least squared problem to estimate the values of the nonzero elements.

$$\mathbf{z}_t = \arg\min_{\mathbf{z}} \left\| \mathbf{y} - \mathbf{\Phi}_t \mathbf{z} \right\|_2$$

5) Calculate the compressed measurement signal estimated by the current $\mathbf{z}_t$ as $\mathbf{a}_t$. Then find the residual, $\mathbf{r}_t$.

$$\mathbf{a}_t = \mathbf{\Phi}_t \mathbf{z}_t$$
$$\mathbf{r}_t = \mathbf{y} - \mathbf{a}_t$$

6) Go to Step *1)* to select the remaining non-zero elements

The selection of non-zero elements provides the indices of non-zero elements and their values in Λ_k and $\mathbf{z}_k$, respectively. The sparse signal, $\hat{\mathbf{x}}$, can be reconstructed as follows.

1) Set every element in $\hat{\mathbf{x}}$ to zero.

2) Set the λ_i^{th} element of $\hat{\mathbf{x}}$ to z_i, where $i = [1,k]$ and z_i is the i^{th} element of $\mathbf{z}_k$.

Though OMP-PKS is fast and more robust to noise than OMP, it has no mechanism against impulsive noise. Therefore, it fails to reconstruct from $\mathbf{y}$ corrupted by impulsive noise.

2.3 Reconstruction in Impulsive Noise Environment

The robustness of the CS reconstruction in this proposal is defined as the tolerance to impulsive noise. There are two major approaches in the robust CS reconstruction: (1) a robust statistic [17-19] and (2) a signal model approaches [20].

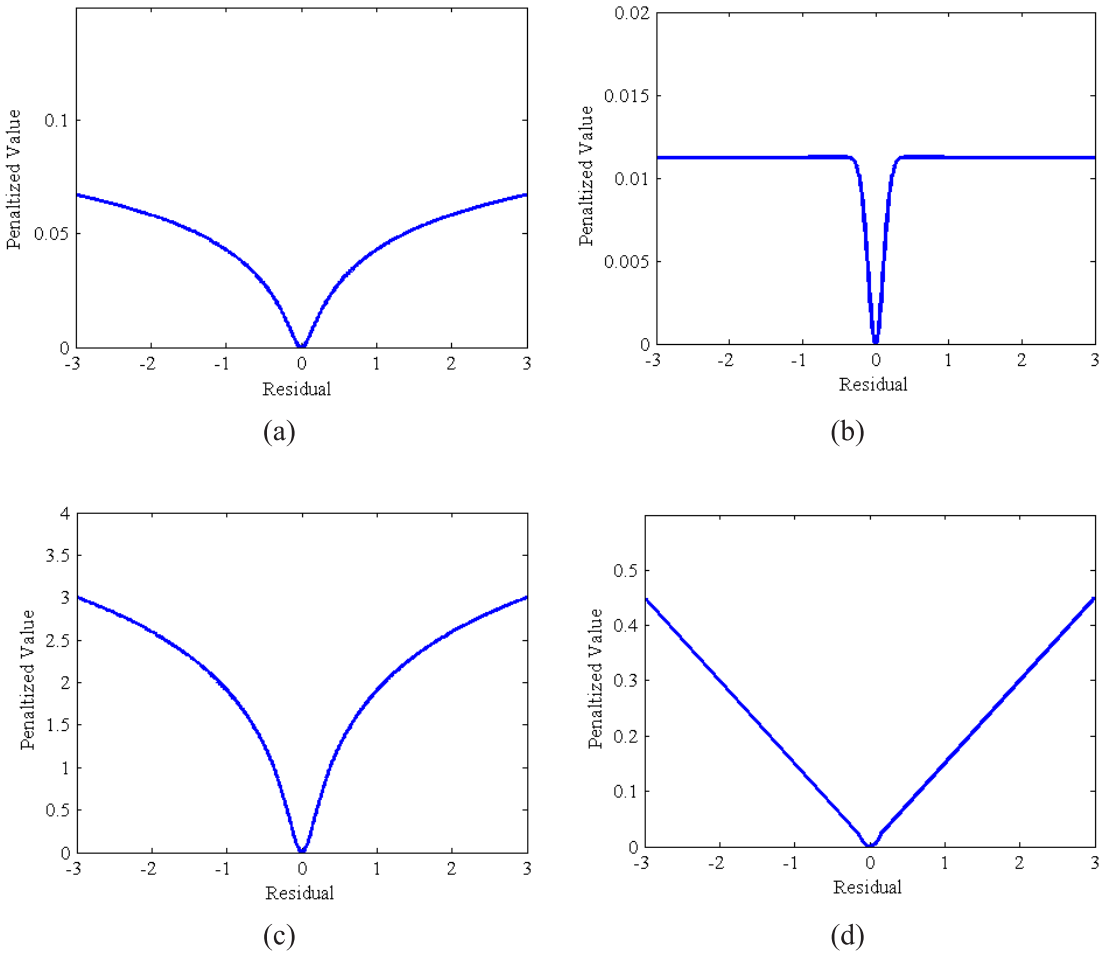


Figure 2.1: Examples of function in robust statistics. (a) Cauchy, (b) Welsh, (c) Lorentzian and (d) Huber functions.

2.3.1 Robust Statistics Approach

When y was corrupted by noise, the noise variance affects the quality of a reconstructed signal. In conventional CS reconstruction techniques, the reconstruction error is assumed to be bounded within a limit and penalized as an L_2 norm. The L_2 norm as a penalty function is effective when y was corrupted by noise with small variance. When y was corrupted by noise with large variance, such as impulsive noise, the large penalty by L_2 norm leads to a failed reconstruction. L_2 norm is replaced with a function in robust statistics for the reconstruction of signal corrupted by noise with large variance.

Figure 2.1 shows examples of the function of robust statistics. The value of the most functions in the figure is bounded; thus, it is less affected by large error (x-axis value). However, the bounded function is not convex, so it requires additional mechanism to ensure the local minimum. In case of the Huber function (Figure 2.1(d)), the value is not bounded; however, it has slower increasing rate after the error is larger than 1; therefore, the effect of the large noise is less than L_2 norm.

i. Lorentzian Function

A generalized Cauchy distribution (GCD) is often used as a model of impulsive noise probability density function. M-GC estimator is an estimator for reconstructing signals that are corrupted by noise in GCD family. M-GC estimator is a local estimator based on Maximum Likelihood (ML). The penalty function of M-GC estimator is the LL_p norm (ρ) defined as follows.

$$\rho(u) = \log\left\{a\gamma\left(\gamma^p + |u|^p\right)^{-2/p}\right\} \tag{2.6}$$

where $a = p\Gamma(2/p)/2(\Gamma(1/p))^2$, where $\Gamma(\cdot)$ is the Gamma function defined as follows;

$$\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt \tag{2.7}$$

$u \in \mathbf{R}^n$ is the reconstruction error;

γ is the scale parameters;

p is the tail constant.

Lorentzian function is the special case of Equation (2.6), where p is set to 2. The Lorentzian function is used as the penalty function in Lorentzian Basis Pursuit (LBP) [12]. In LBP, CS reconstruction is formulated as follows.

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ s.t. } g_\gamma(\mathbf{y} - \mathbf{\Phi x}) \le \varepsilon, \tag{2.8}$$

where $g_\gamma(\cdot)$ is the Lorentzian penalty function and is defined as

$$g_\gamma(\mathbf{a}) = \sum_{i=1}^{n} l_\gamma(a_i), \tag{2.9}$$

where a_i is the i^{th} element of $\mathbf{a} \in \mathbf{R}^n$;

$l_\gamma(\cdot)$ is the Lorentzian function and is defined as follows.

$$l_\gamma(a_i) = \log\left(1 + \frac{|a_i|^2}{\gamma^2}\right). \tag{2.10}$$

$\gamma > 0$ is defined according to the noise variance which is *a priori* in [17]. The magnitude of Lorentzian function increases very slowly, after the magnitude of a_i reaches γ, hence, the effect of impulsive noise can be reduced. According to [17], γ is set as the median absolute deviation (MAD) of $\mathbf{y}–\mathbf{\Phi x}$.

Though the reconstructed sparse signal, $\hat{\mathbf{x}}$, has the error bounded by the Lorentzian function, it may not be optimal. Consequently, the post-processing in form of the regression is applied to ensure that non-zero elements in $\hat{\mathbf{x}}$ have large magnitude. The regression is formulated as the following problem.

$$\tilde{\mathbf{x}}_I = \arg\min_{\mathbf{x} \in \mathbf{R}^d} g_\xi(\mathbf{y} - \mathbf{\Phi}_I \mathbf{x}) \tag{2.11}$$

where $I = \{i; |\hat{x}_i| > \alpha\}$,where $\alpha > 0$ is the constant (to be determined later); $\hat{x}_i$ is the i^{th} element of $\hat{\mathbf{x}}$ and $|\cdot|$ is the magnitude; I is the set containing the indices of

non-zero elements;

d is the size of I;

$\tilde{\mathbf{x}}_I \in \mathbf{R}^d$ is the vector containing the value of the nonzero elements;

$\mathbf{\Phi}_I \in \mathbf{R}^{d \times M}$ is the random measurement matrix after every column whose index is not in I is removed;

ξ is the scale parameter for noise distribution.

α is used to select the non-zero elements in the reconstruction by Equation (2.8). From the experiment in [17], the optimal value of α is $\lambda \max_i |\hat{x}_i|$ where $0 < \lambda < 1$.

After Equation (2.11) is applied, the sparse signal, $\hat{\mathbf{x}}$, is reconstructed as follows.

1) Set every element in $\hat{\mathbf{x}}$ to zero.

2) Set the I_i^{th} element of $\hat{\mathbf{x}}$ to $\tilde{x}_i$ where $i = [1, k]$; I_i and $\tilde{x}_i$ are the i-th elements of I and $\tilde{\mathbf{x}}_i$, respectively.

Because the Lorentzian function is not convex, the global minimum is not guaranteed. Complex optimization and parameter adjustments are often required. To reduce the computational complexity, Lorentzian Iterative Hard Thresholding (LIHT) [18] is proposed as a greedy alternative of [17]. The sparse representation is approximated by the hard thresholding. The reconstruction in LIHT is according to the following equation.

$$\arg\min_{\mathbf{x}} g_\gamma(\mathbf{y} - \mathbf{\Phi x}) \quad \text{s.t.} \quad \|\mathbf{x}\|_0 \le k, \tag{2.12}$$

where $g_\gamma(\mathbf{y} - \mathbf{\Phi x})$ is defined as Equation (2.9). The algorithm of LIHT is as follows.

1) Set $\hat{\mathbf{x}}(0)$ and t to zero vector and 0, respectively.

2) Set $\hat{\mathbf{x}}(t+1)$ as follows.

$$\hat{\mathbf{x}}(t+1) = \mathbf{H}_k(\hat{\mathbf{x}}(t) + \mu\mathbf{S}(t)), \tag{2.13}$$

where $\mathbf{H}_k(\cdot)$ is the non-linear operator, which keeps only k largest components, and sets the remaining components to zero. μ is the step size. $\mathbf{S}$ is defined as follows.

$$\mathbf{S}(t) = \mathbf{\Phi}^T \mathbf{W}_t(\mathbf{y} - \mathbf{\Phi}\hat{\mathbf{x}}(t)), \tag{2.14}$$

where $\mathbf{W}_t \in \mathbf{R}^{M\times N}$ is an diagonal matrix whose diagonal elements are defined as follows.

$$\mathbf{W}_t(i,i) = \frac{\gamma^2}{\gamma^2 + (y_i - \mathbf{\Phi}_i^T \hat{x}(t))^2}, i = 1,....,M. \tag{2.15}$$

The step size is set according to the following equation.

$$\mu(t) = \frac{\left\|\mathbf{S}_{k(t)}(t)\right\|_2^2}{\left\|\mathbf{W}_t^{1/2}\mathbf{\Phi}_{k(t)}\mathbf{S}_{k(t)}(t)\right\|_2^2}. \tag{2.16}$$

When $g_\gamma(\mathbf{y} - \mathbf{\Phi}\hat{\mathbf{x}}(t+1)) > g_\gamma(\mathbf{y} - \mathbf{\Phi}\hat{\mathbf{x}}(t))$, $\mu(t)$ is set to $0.5\mu(t)$.

3) Terminate when $\mathbf{\Phi}\hat{\mathbf{x}} - \mathbf{y}$ is within a predefined error bound.

Although LIHT has the benefit over LBP in that it is fast and robust, it shares the same problem as IHT. It requires that $\mathbf{x}$ must be very sparse or that the compression has high measurement rate. Though IHT is faster than OMP, it is less stable.

ii. Huber function

Huber function has a benefit over Lorentzian function in that it is convex. The CS reconstruction using Huber function is formulated as the following convex optimization problem [19].

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} h(\mathbf{y} - \mathbf{\Phi}\mathbf{x}) + \alpha\left\|\mathbf{x}\right\|_1, \tag{2.17}$$

where $h(\cdot)$ is the Huber penalty function and defined as follows.

$$h(\mathbf{a}) = \sum_{i=1}^{n} \rho(a_i), \tag{2.18}$$

where a_i is the i^{th} element of $\mathbf{a} \in \mathbf{R}^n$;

$\rho(\cdot)$ is the Huber function and is defined as

$$\rho(a_i) = \begin{cases} a_i^2 & ;|a_i| \le 1 \\ 2a_i - 1 & ;|a_i| > 1 \end{cases}. \tag{2.19}$$

The optimum value of α is proved in [19] to be between 0 and $\alpha_{\max}$ which is defined as

$$\alpha_{\max} = \left\| \mathbf{\Phi}^T \boldsymbol{\rho}(\mathbf{y}) \right\|_{\infty}, \tag{2.20}$$

where $\boldsymbol{\rho}(\mathbf{y}) = [\rho(y_1)\ \rho(y_2)\ \ ...\rho(y_M)]^T$ and $\|\cdot\|_{\infty}$ is the L_∞ norm.

Equation (2.17) can be optimized by many existing convex optimization solvers. The global minimum solution is guaranteed. However, Huber function is not bounded; therefore, when the noise is large and/or dense, the quality of the reconstruction is degraded.

2.3.2 Signal Model Approach

Sermwuthisarn et. al. [20] proposed the detection and the removal of impulsive noise by the characteristic of **x.** When **x** is an image sparsified by octave-tree discrete wavelet transform (DWT), most of its energy is confined in the third level (L_3) subband. However, when **x** is reconstructed from **y** contaminated by impulsive noise, lots of its energy will leak out of L_3 subband. Figure 2.2 shows two examples of the reconstruction from the noisy **y** by OMP-PKS [12]. OMP-PKS was used because it has no mechanism against impulsive noise so the effect of noise is clearly shown. Compared to the original **x**'s (Figures 2.2(a) and 2.2(c)), the reconstruction results from the noisy **y**'s (Figures 2.2(b) and 2.2(d)) had large energy leaking out of L_3 subband (right of the red line).

Though it is impossible to detect impulsive noise directly from **y**, Figure 2.2 indicates that the presence of noise can be detected via the energy leaking out of L_3 subband in the reconstructed $\hat{\mathbf{x}}$. The method in [20], namely Approximated Measurement Preprocessing (AMP) in this thesis, iteratively removes the elements in **y** until most of the energy in $\hat{\mathbf{x}}$ is confined inside L_3 subband. The removal order is from the maximum to the minimum magnitudes. The energy leaking out of L_3 subband is measured as the ratio to the total energy. The ratio is referred in this thesis as the Energy Ratio (*ER*) and is defined as follows.

$$ER = \frac{\sum_{i \notin L_3} \hat{x}_i^{\,2}}{\sum_{j=1}^{N} \hat{x}_j^{\,2}}, \tag{2.21}$$

where $\hat{x}_i$ and L_3 are the i^{th} element of $\hat{\mathbf{x}}$ and the set of the wavelet coefficients in the third level subband.

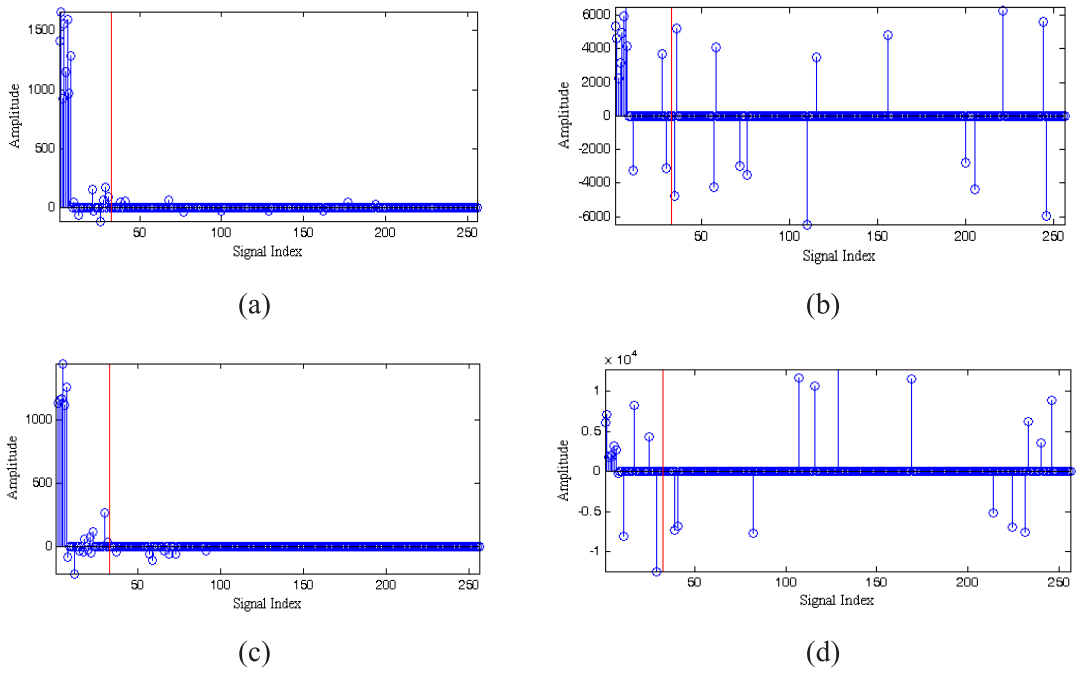


Figure 2.2: The effect of impulsive noise to the CS reconstruction. Figures (a) and (c) are the original sparse signals; Figures (b) and (d) are the reconstructed sparse signals for the signal shown in Figures (a) and (c), respectively. The elements at the left of the red line are inside the L_3 subband.

After the noise corrupted elements are all removed, the removed elements will be approximated from the remaining noiseless elements. AMP consists of 2 stages: the removal and the approximation of the noise corrupted elements. The removal of the noise corrupted elements in AMP is as follows.

1) Apply OMP-PKS to reconstruct the sparse signal, $\hat{\mathbf{x}}$, from the compressed measurement signal, $\mathbf{y}$.

2) Determine whether there is impulsive noise in $\mathbf{y}$. If the *ER* of $\hat{\mathbf{x}}$ in Step *1*) is less than the predefined energy ratio threshold (η), AMP is terminated.

3) Remove the elements with the largest magnitude in $\mathbf{y}$. Note that more than one element may be removed in this step.

4) Apply OMP-PKS to reconstruct $\hat{\mathbf{x}}$ from the remaining $\mathbf{y}$.

5) Terminate the removal process if *ER* of $\hat{\mathbf{x}}$ in *4)* is lower than η. Otherwise, go to Step *3)*.

The removal of elements will not lead to very low possibility of successful reconstruction, if the removal amount is within 40% of the size of $\mathbf{y}$ [20]. $\mathbf{y}$ after the removal stage, defined as $\mathbf{y}_s$, contains only the noise-free elements. In the second stage, the values of the noise corrupted elements are then approximated from $\mathbf{y}_s$. The algorithm is as follows.

1) Apply OMP-PKS to reconstruct the approximated sparse signal, $\hat{\mathbf{x}}_s$, from $\mathbf{y}_s$.

2) Set $\tilde{\mathbf{y}}$ to $\mathbf{\Phi}\hat{\mathbf{x}}_s$.

3) Set the estimated noise-free $\mathbf{y}$, $\mathbf{y}_{nf}$, according to the following equation.

$$y_{nf,i} = \begin{cases} \tilde{y}_i & ; y_i \text{ is the noise corrupted element.} \\ y_i & ; \text{otherwise} \end{cases}, \tag{2.22}$$

where $y_{nf,i}$, $\tilde{y}_i$ and y_i are the i^{th} elements of $\mathbf{y}_{nf}$, $\tilde{\mathbf{y}}$ and $\mathbf{y}$, respectively.

The threshold in AMP can be easily set by experiment. AMP requires no complex optimization. Furthermore, it is robust against very large noise. However, it requires n+1 reconstruction when $\mathbf{y}$ is corrupted by n levels of impulsive noise. Even though OMP-PKS is a greedy reconstruction, the application of OMP-PKS for n+1 times leads to very high computational time.

CHAPTER III

PROPOSED METHODS

This chapter addresses the reconstruction of a compressed measurement signal (**y**) in impulsive noise environments. Two preprocessing algorithms are proposed in this dissertation. Section 3.1 introduces a basic idea of how to reduce the computational time of AMP. Sections 3.2 and 3.3 describe two proposed preprocessing algorithms: Greedy Boundary Finder (GBF) and Greedy Steep Slope Finder (GSSF). In GBF and GSSF a binary search is adopted to approximate the number of noisy elements to within $+g$ of the actual number, where g is the predefined constant and has the unit of percent of the size of **y**.

3.1 Basic Idea

When an image is sparsified by octave-tree discrete wavelet transform (DWT), most of its energy is confined in the third level (L_3) subband. In Approximated Measurement Preprocessing (AMP) [20], the amount of the energy leaking out of the L_3 subband is used as the indicator of the presence of impulsive noise. AMP is divided into two stages: (1) the noise removal and (2) the approximation stages. The approximation stage is applied after all noise has been removed in noise removal stage. The removal without any approximation of the removed elements in-between is possible, because information in an image is highly redundant. In this thesis, the information redundancy is further exploited to reduce the computational time of AMP. It is hypothesized that successful CS reconstruction is possible even when some largest elements in **y** is missing. Therefore, it is unnecessary to remove only the noise contaminated elements as in AMP. Binary search is applied to estimate the number of the noisy element to within the predefined range of the correct value.

Figures 3.1 - 3.3 show three examples of the reconstruction from the incomplete **y**, when **y** was corrupted by impulsive noise. In all three images, Figure

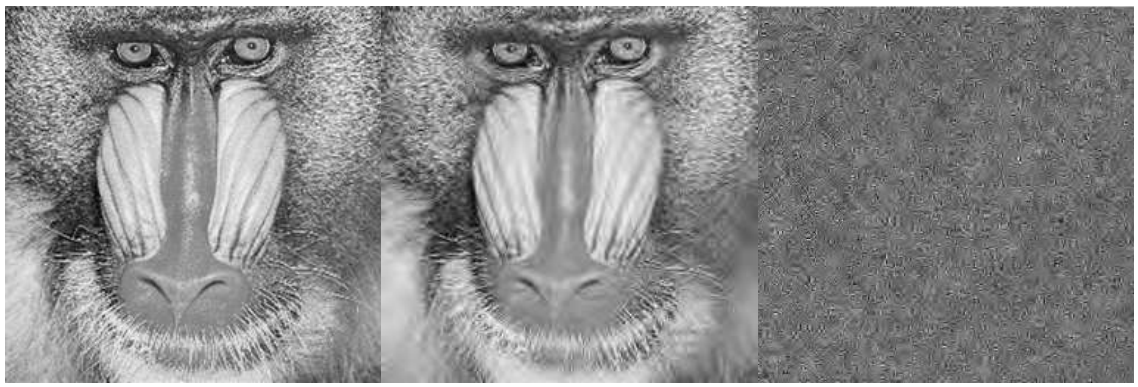
(a) shows the original images. Figure (b) shows the image after it was sparsified by wavelet shrinkage thresholding [22]. When **y** was corrupted by impulsive noise, Figures (c) - (f) show the reconstruction by OMP-PKS. Figure (c) shows the reconstruction when every element in **y** (both noisy and noise-free elements) was used. The figure clearly indicates that OMP-PKS failed to reconstruct the image when the elements corrupted by impulsive noise were included. Figure (d) shows the reconstruction when all noisy elements were removed; Figure (e) shows the reconstruction when the additional noiseless elements (4.5%, 4.5% and 4.6% of **y** in Figures 3.1, 3.2 and 3.3, respectively) were also removed. Figures (b), (d) and (e) were almost similar; therefore, the degradation was due to the wavelet shrinkage thresholding. These results supported the hypothesis that some largest elements could be removed without severe visual degradation. However, if too many elements were removed as shown in Figure (f) (14.5%, 14.5% and 14.5% of **y** in Figures 3.1, 3.2 and 3.3, respectively), the visual degradation would be distinct. From Figures 3.1 - 3.3, it could be concluded that though the largest element could be removed, the number of removed elements should not be high

Though the visual quality of Figure (f) was distinctly lower than Figure (e), the image still conveyed most information of the original image. It implied that the structure of the signal might be conserved. In the contrary, the information of Figure (c) was almost entirely destroyed and the structure of the signal in term of the energy distribution was changed [20].

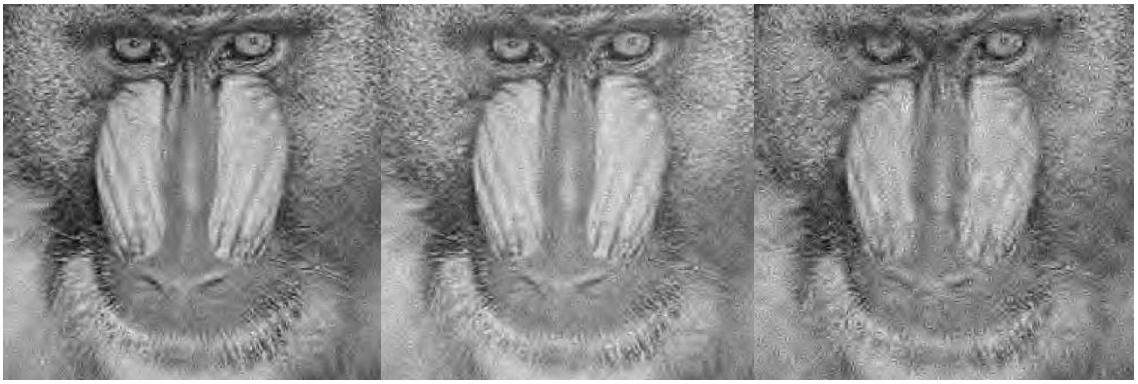
(a) (b) (c)

(d) (e) (f)

Figure 3.1: The reconstruction of Peppers when 5.5% of **y** was contaminated by impulsive noise. (a) original image; (b) image after wavelet shrinkage thresholding; the reconstructed image when (c) 0%, (d) 5.5% , (e) 10% (the number of removed noiseless elements = 4.5% of **y**) and (f) 20% (the number of removed noiseless elements = 14.5% of **y**) of **y** was removed. The removal order is from the maximum to the minimum magnitudes.



(a) (b) (c)



(d) (e) (f)

Figure 3.2: The reconstruction of Mandrill when 5.5% of **y** was contaminated by impulsive noise. (a) original image; (b) image after wavelet shrinkage thresholding; the reconstructed image when (c) 0%, (d) 5.5% , (e) 10% (the number of removed noiseless elements = 4.5% of **y**) and (f) 20% of **y** was removed (the number of removed noiseless elements = 14.5% of **y**). The removal order is from the maximum to the minimum magnitudes.

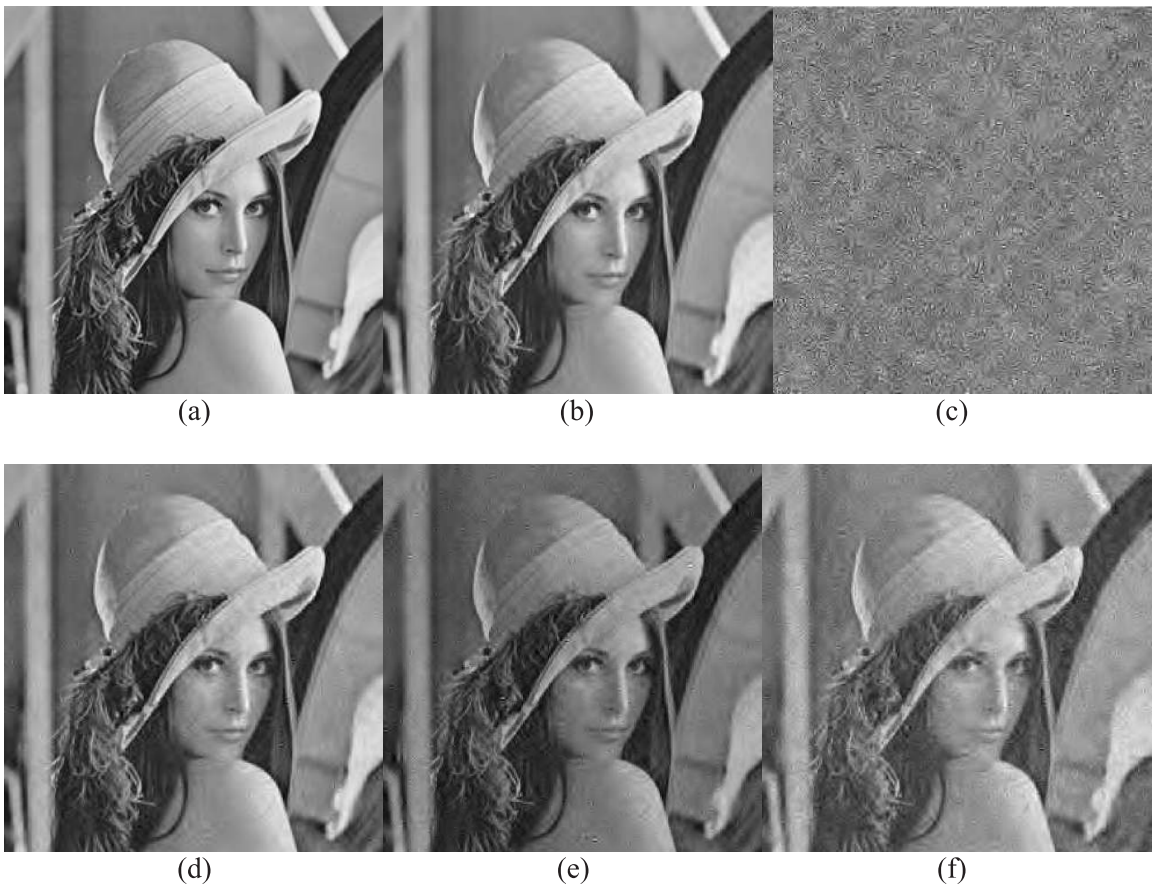
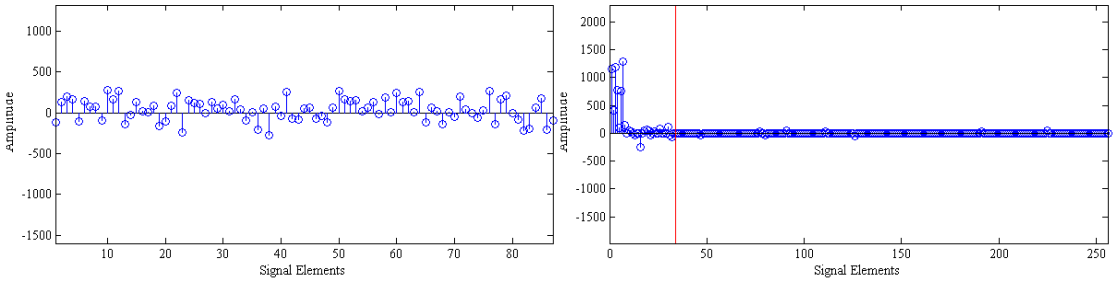


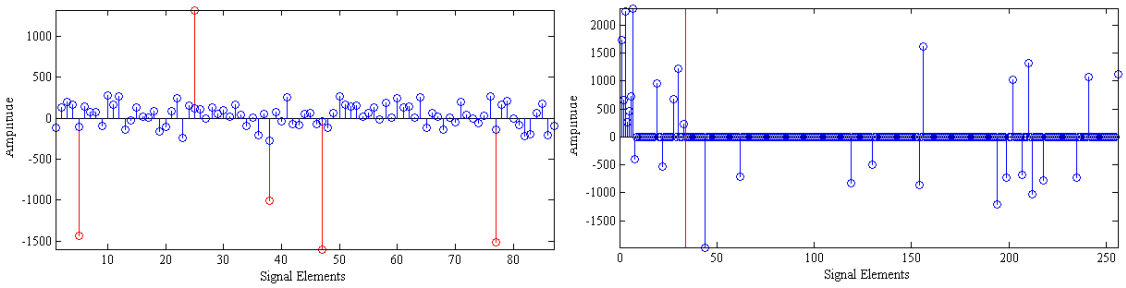
Figure 3.3: The reconstruction of Lena when 5.4% of **y** was contaminated by impulsive noise. (a) original image; (b) image after wavelet shrinkage thresholding; the reconstructed image when (c) 0%, (d) 5.4% , (e) 10% (the number of removed noiseless elements = 4.6% of **y**) and (f) 20% (the number of removed noiseless elements = 14.6% of **y**) of **y** was removed. The removal order is from the maximum to the minimum magnitudes.

Figures 3.4 - 3.8 show one example of how the energy distribution changed in the reconstruction from **y** corrupted by impulsive noise. In every plots, Figures (a) and (b) show **y** and the reconstructed $\hat{\mathbf{x}}$, respectively. The elements corrupted by impulsive noise are depicted in red in Figure (a). The elements inside the L_3 subband were located at the left of the red line in Figure (b). Figure 3.4 shows the original signal. When the noisy element in **y** was included in the reconstruction (Figure 3.5), the energy distribution of $\hat{\mathbf{x}}$ was considerably changed. Lots of energy was leaked out of L_3 subband. On the other hand, when the noisy elements were not included (Figures 3.6 - 3.8), most energy of $\hat{\mathbf{x}}$ was confined within the L_3 subband The energy distribution characteristic held even when 14.5% of the largest noiseless element were not used in the reconstruction (Figure 3.8).



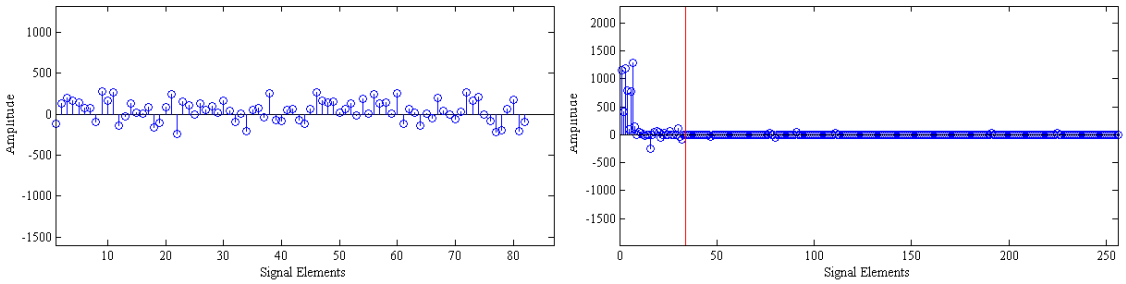
(a) (b)

Figure 3.4: Original signal: (a) compresed measurement signal (**y**) and (b) the sparse signal ($\hat{\mathbf{x}}$). The left of the red line in Figure (b) indicates the region in L_3 subband.



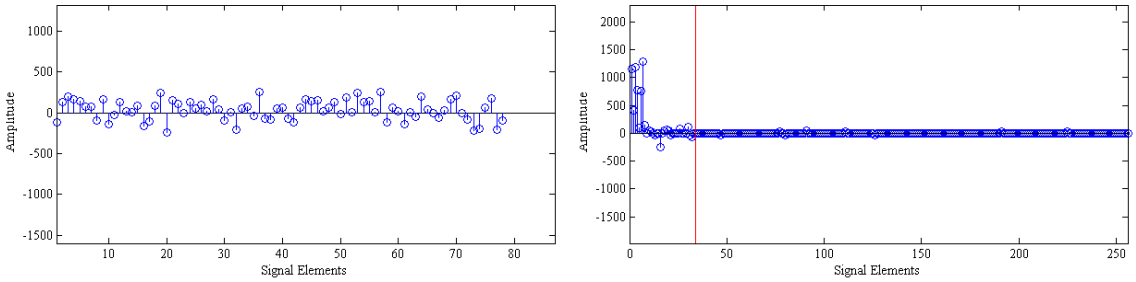
(a) (b)

Figure 3.5: The reconstruction of **y** corrupted by impulsive noises, when every element in **y** was used: (a) **y** used in the reconstruction and (b) its corresponding $\hat{\mathbf{x}}$. Impulsive noise corrupted element is depicted in red in Figure (a). The left of the red line in Figure (b) indicates the region in L_3 subband.



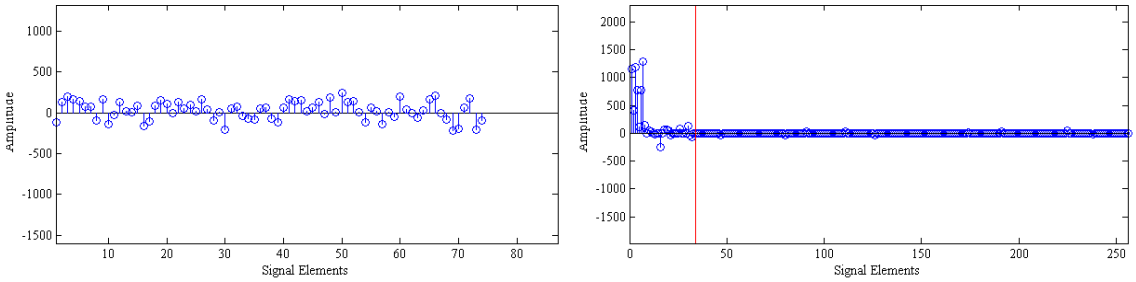
(a) (b)

Figure 3.6: The reconstruction of **y** when only the noiseless elements in **y** are used: (a) **y** used in the reconstruction and (b) its corresponding $\hat{\mathbf{x}}$. The left of the red line in Figure (b) indicates the region in L_3 subband.



(a) (b)

Figure 3.7: The reconstruction example when the highest 10% of **y** were removed (the number of noisy elements = 5.5% of **y**, and the number of removed noiseless elements = 4.5% of **y**): (a) **y** used in the reconstruction and (b) its corresponding $\hat{\mathbf{x}}$. The left of the red line in Figure (b) indicates the region in L_3 subband.



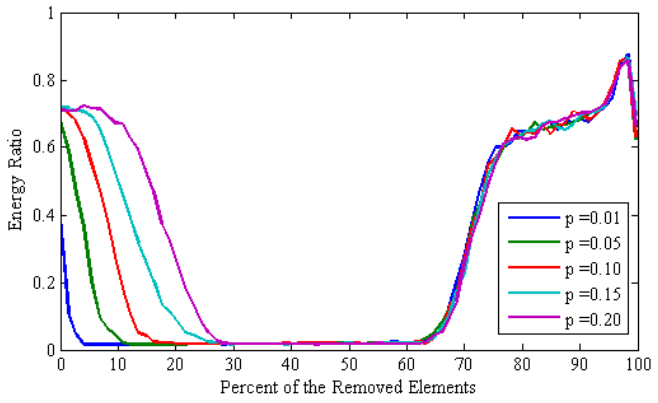
(a) (b)

Figure 3.8: The reconstruction example when the highest 20% of **y** were removed (the number of noisy elements = 5.5% of **y**, and the number of removed noiseless elements = 14.5% of **y**): (a) **y** used in the reconstruction and (b) its corresponding $\hat{\mathbf{x}}$. The left of the red line in Figure (b) indicates the region in L_3 subband.

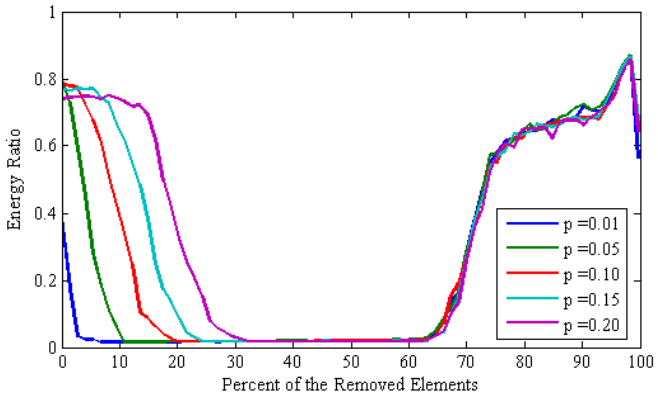
Since the energy distribution does not change when the incomplete **y** is used, it can be used to indicate whether the reconstruction is successful. The energy distribution is measured by (leakage) energy ratio (*ER*, Equation (2.21)). The effect of the number of the removed element to *ER* is investigated. Figure 3.9 shows two examples when **y** was contaminated by impulsive noise. A point in the graph was the average from 100 reconstructed images of 16×16 pixels. Different colors depict different impulsive noise densities (in term of the noise probability, p). The removal order was from the maximum to the minimum magnitudes. The measurement rate, the ratio between the length of **y** and the length of **x** (M/N), was set to 0.3.

Figures 3.9(a) and 3.9(b) show the relationship of *ER* and the percentage of removed elements at the noise magnitude means (μ) of $5y_{max}$ and $10y_{max}$, respectively.

At the left and the right of the graphs, there were the regions with high *ER*, the deviation from the general characteristic of an image. In the left of the graph, the number of removed elements that the *ER* dropped to the low ratio was higher when p was larger, and the magnitude of *ER* depended on μ. The dependence of the noise parameters indicated that the high *ER* at the left was caused by noise. The regions with the high *ER* at the right were almost the same for all p's and μ's. The independence of the noise parameters indicated that the high *ER* was not caused by noise but by the lack of information in $\mathbf{y}$ to reconstruct $\hat{\mathbf{x}}$. The low *ER* at the middle of the graphs corresponded to the reconstruction in (d) - (f) of Figures 3.1 - 3.3. The distribution complied with the image characteristic and indicated the high probability of successful reconstruction of $\hat{\mathbf{x}}$, i.e. the impulsive noise was completely removed and there was sufficient information left in $\mathbf{y}$ for reconstruction.



(a)



(b)

Figure 3.9: The relationship between the (leakage) energy ratio and the percent of the removed elements in the compressed measurement at the measurement rate of 0.3. The magnitude mean of impulsive noise were (a) $5y_{max}$ and (b) $10y_{max}$.

Since the complete removal of impulsive noise corrupted elements led to the drop in *ER* at the middle of the graphs in Figure 3.9, the impulsive noise rejection is the problem of finding the left border of the region with low *ER*. Binary search, which is one of efficient search strategies with O(log(*n*)) cost, is adopted in this thesis. The lower and the upper boundaries of the binary search are the minimum and the maximum numbers of removed elements, respectively. The lower boundary (α) is initialized as zero (every element is used), while the upper boundary (β) is initialized as 40% of the size of **y** (0.4*M*). The value of 40% is taken from [20]. Thus, it is assumed that at most 40% of **y** is corrupted by impulsive noise. The cost function is applied to calculate the cost of removing the $0.5(\alpha+\beta)$ highest elements from **y** (the middle of the upper and the lower boundaries). α and β are then updated according to the cost function. The binary search is applied until the difference between α and β is less than the threshold, which indicates that the difference to the actual number of noisy elements is within the desired resolution. The threshold is defined as the gap resolution (*g*) with the unit of percent to the size of *y*.

In this thesis, two cost functions are proposed. In the first cost function, the region of low *ER* is defined as the "gap" at the middle of the graph. The value of *ER* is directly used to search for the minimum number of removed elements that is still in the gap. The preprocessing using the first cost function is defined as Greedy Boundary Finder (GBF) and is described in Section 3.2.

In the second cost function, the left border of the gap is detected as the abrupt change in *ER*. Average slope is used to estimate the amount of change. The preprocessing using the second cost function is defined as Greedy Steep Slope Finder (GSSF) and is described in Section 3.3.

3.2 Greedy Boundary Finder (GBF)

In GBF, *ER* at $0.5(\alpha+\beta)$ is directly used to detect whether it is within the gap or not. If the *ER* is low, it indicates that all impulsive noise is rejected; thus, $0.5(\alpha+\beta)$ is the maximum possible number of noise corrupted elements; β, which is the maximum number of noisy (removed) elements, is changed to $0.5(\alpha+\beta)$. In the

contrary, the high *ER* at $0.5(\alpha+\beta)$ indicates that the incomplete removal of noisy elements; therefore, α, which is the minimum number of noisy elements, is changed to $0.5(\alpha+\beta)$. The algorithm of GBF is as follows.

1) Apply OMP-PKS to reconstruct the sparse signal, $\hat{\mathbf{x}}$, from the input measurement signal, $\mathbf{y}$.

2) Determine whether there is impulsive noise in $\mathbf{y}$. If *ER* of $\hat{\mathbf{x}}$ in Step *1)* is less than the predefined energy ratio threshold (η), GBF is terminated. *ER* is calculated according to Equation (2.21).

3) Set α and β to 0% and 40%, respectively. 40% is used as the upper removal boundary, because it is the highest number of noisy elements in the assumption.

4) Set ς to $0.5(\alpha + \beta)$. Define $\mathbf{y}_{\varsigma}$ as $\mathbf{y}$ after the highest ς elements (according to the magnitude) have been removed.

5) Apply OMP-PKS to reconstruct the sparse signal, $\hat{\mathbf{x}}_{\varsigma}$, from $\mathbf{y}_{\varsigma}$.

6) Update the upper and lower boundaries. If *ER* of $\hat{\mathbf{x}}_{\varsigma}$ is less than η (no impulsive noise in $\mathbf{y}_{\varsigma}$), set β to ς; otherwise set α to ς.

7) Go to Step *4)*, if the difference between α and β is more than the gap resolution, g.

8) Set $\mathbf{y}_s$ to $\mathbf{y}$ after the highest β elements (according to the magnitude) have been removed. The highest β elements are considered as the noisy elements. $\mathbf{y}_s$ is considered as the impulsive-noise-free signal.

In the same manner as AMP, the removal stage in GBF is followed by the approximation stage to estimate the values of the removed elements. The method in AMP (Section 2.3.2) is adopted.

Table 3.1: The number of reconstruction in GBF and AMP

The number of noise level	**The number of reconstruction**					
	GBF (g=20%)	***GBF (g=10%)***	***GBF (g=5%)***	***GBF (g=2%)***	***GBF (g=1%)***	***AMP***
pM[a]	2	3	4	5	6	pM+1

a. p and M are the probability of impulsive noise and the size of **y**, respectively.

Since most of the computational load in GBF belongs to the reconstruction process, the number of reconstruction is kept constant by having the value of g in the unit of the percent of the size of **y**. g is set as the ratio to the initial value of β which is 40%; g can be 20%, 10%, 5%, 2.5%, 1.25%, …. For simplicity, g is set to the highest integer that is lower than the possible gap resolution. Table 3.1 compares the number of reconstruction required by AMP and GBF at various g.

3.3 Greedy Steep Slope Finder (GSSF)

The absolute threshold, η, is used to distinguish the region of low *ER* in GBF. However, it is not guaranteed the reconstruction from the noiseless elements always leads to $\hat{\mathbf{x}}$ whose *ER* is smaller than η. If the reconstruction with noiseless **y** has *ER* larger than η, the incorrect update of α and β in GBF will lead to an excess removal and lots of noise-free elements (useful information) will be lost. Since it is possible to detect the change of *ER* as the indicator of complete removal of impulsive noise, the slope of *ER* between $0.5(\alpha+\beta)$ and β is used to detect the complete noise removal in GSSF.

If the slope is steep, it indicates the complete removal of the noisy elements. However, the left border of the gap becomes steeper when the number of removed elements is smaller. Therefore, the detection for the steepest slope at any particular point will lead to the rightmost point of the high *ER* region. In GSSF, the steep slope is detected indirectly via the slope to the point where *ER* is low. If the slope is small, it implies all noisy elements are removed and the steep slope is to the left of the current point. Since the steepness of the slope (the left border of the gap in Figure 3.9) depends

on the noise parameters, the threshold to detect the shallow slope is estimated as follows.

$$T = \frac{c(E_{40} - E_0)}{40}, \tag{3.1}$$

where T is the slope threshold;

c is the predefined slope scale;

E_{40} is the *ER* of $\hat{\mathbf{x}}$ reconstructed from $\mathbf{y}$ after its largest 40% elements are removed;

E_0 is the *ER* of $\hat{\mathbf{x}}$ reconstructed from $\mathbf{y}$ when every element is used.

Similar to GBF, the process of GSSF is divided into two stages: (1) noise removal and (2) approximation stages. OMP-PKS is adopted as the reconstruction algorithm in GSSF. The process in the noise removal stage is as follows.

1) Initialize α and β to 0% and 40%, respectively.

2) Determine whether there is impulsive noise in $\mathbf{y}$. If E_α is less than E_β, GSF is terminated since there is no impulsive noise in $\mathbf{y}$.

3) Calculate the slope threshold (T) according to (3.1).

4) Set ξ to $0.5(\alpha + \beta)$.

5) Calculate the estimated slope (δ) between ξ and β as follows.

$$\delta = \frac{(E_\xi - E_\beta)}{(\beta - \xi)}$$

E_ξ and E_β are the *ER* of $\hat{\mathbf{x}}$ reconstructed from $\mathbf{y}$ after its largest ξ% and β% elements are removed, respectively.

6) Update α and β. If δ is less than T, set β to ξ; otherwise set α to ξ

7) Go to Step *4)*, if the difference between α and β is more than g.

Table 3.2: The number of reconstruction used by GSSF and AMP

The number of noise level	**The number of reconstruction**				
	GSSF (g=10%)	***GSSF (g=5%)***	***GSSF (g=2%)***	***GSSF (g=1%)***	***AMP***
pM^*	4	5	6	7	pM+1

* p and M are the probability of impulsive noise and the size of **y**, respectively.

8) Set $\mathbf{y}_s$ to **y** after its largest β elements (according to the magnitude) have been removed. The largest β elements are considered as the noisy elements and $\mathbf{y}_s$ is the noiseless signal.

The noise removal stage is followed by the approximation stage. The algorithm in AMP is adopted.

In the same reasoning as GBF, the number of reconstruction in GSSF is kept constant by having the value of g in the unit of the percent of the size of **y**. Table 3.2 compares the number of reconstruction required by AMP and GSSF at various g.

CHAPTER IV

EXPERIMENT AND DISCUSSION

The proposed algorithms are evaluated in this chapter. Experiment setup is first presented in Section 4.1. Greedy Boundary Finder (GBF) and Greedy Steep Slope Finder (GSSF) are evaluated in Sections 4.2 and 4.3, respectively. The comparison between GBF and GSSF is presented in Section 4.4. Finally, the limitation of the (leakage) energy ratio (*ER*, Equation (2.21)) as the mean for detecting the impulsive noise in compressed measurement signal (**y**) is provided in Section 4.5.

4.1 Experiment Setup

The experiment was conducted on a PC with 2.83 GHz Intel Core 2 Quad CPU and 4 GB of RAM. All methods were implemented by 64-bit MATLAB R2011a. Two datasets were used in the experiment: (1) 100 image blocks of the size 16×16 pixels and (2) 20 test images (Figure 4.1). Images in the first row and the second row are the standard test images. The remaining images in the third row and the fourth row are the artificial images. (The artificial images are available at http://sourceforge.net/projects/testimages/files/.) All test images were resized to 256×256 pixels.

Octave-tree DWT with db8 as the mother wavelet was used to transform images to sparse domain. Wavelet shrinkage thresholding [22] was used to further sparsify the data. Hadamard matrix was used as the random measurement matrix.

In the first dataset, the sparsity rate and the measurement rate (M/N) were set to 0.1 and 0.4 respectively. In the second dataset, each test image was divided into 256 blocks of 1×256 pixels. The sparsity level and the average measurement rate were intentionally varied among blocks to test the tolerance of the algorithms to the

Figure 4.1: The test images used in the experiment.

changes of these two parameters. The average sparsity rate was 0.1. Three average measurement rates (0.30, 0.35 and 0.40) were tested.

The performance of GBF and GSSF were compared with (1) Lorentzian Iterative Hard Thresholding (LIHT) [18], (2) the CS reconstruction using Huber penalty function (HUBER) [19] and (3) Approximated Measurement Preprocessing (AMP) [20]. Since AMP, GBF and GSSF are the preprocessing, Orthogonal Matching Pursuit with Partially Known Support (OMP-PKS) [12] was used to reconstruct the final result.

The reconstruction of the first dataset was evaluated by the computational time and the percent of a mean square error (%*MSE*) which is defined as follows.

$$\%MSE = \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2}{\|\mathbf{x}\|_2} \times 100, \tag{4.1}$$

where $\hat{\mathbf{x}}$ and $\mathbf{x}$ are the reconstructed and the original sparse signals, respectively;

$\|\cdot\|_2$ is the L_2-norm.

The reconstruction of the second dataset was evaluated by PSNR, the computational time and the visual inspection.

4.2 The Evaluation of GBF

The evaluation was divided into two main sections: (1) evaluation for the optimal energy ratio threshold (η) in Section 4.2.1 and (2) the performance evaluation in Section 4.2.2. The first dataset was used in both sections, while the second dataset was used only in Section 4.2.2.

The magnitude of the impulsive noise was set relative to y_{max}, the maximum magnitude of the elements in the original (noiseless) **y**. In Section 4.2.1, the magnitude was fixed to one value. In Section 4.2.2, the distribution of the noise magnitude was Gaussian. The deviation was set to y_{max}. The performance under three different magnitude means ($\mu = 5y_{max}$, $7y_{max}$, and $10y_{max}$) was evaluated. In the second dataset, there were 256 **y**'s for each image, so y_{max} was set to the maximum magnitude among 256 **y**'s. The noise level was described by the probability of the impulsive noise (p). Four levels of p (0.05, 0.10, 0.15, and 0.20) were tested.

4.2.1 Evaluation for the Energy Ratio Threshold (η)

Figure 4.2 shows the relationship between the percent of the inaccurate rejection and energy ratio threshold (η). In the figure, different colors depict different magnitudes of impulsive noise. The range of optimal threshold (low inaccurate rejection) was wider when the magnitude (power) of noise was higher, since the effect of impulsive noise to the *ER* was more distinct. η in the subsequent experiment was set to 0.1, because it was among one of the optimal values for all cases. It should be noted that there were cases when the noiseless **y** would provide the reconstruction results with the *ER* higher than 0.1; in such cases, the preprocessing by GBF and AMP led to the removal of some noiseless elements in **y**. Figure 4.2 also indicates that the detection of the impulsive noise by the *ER* was possible even when the magnitude of impulsive noise was as low as $1.25y_{max}$.

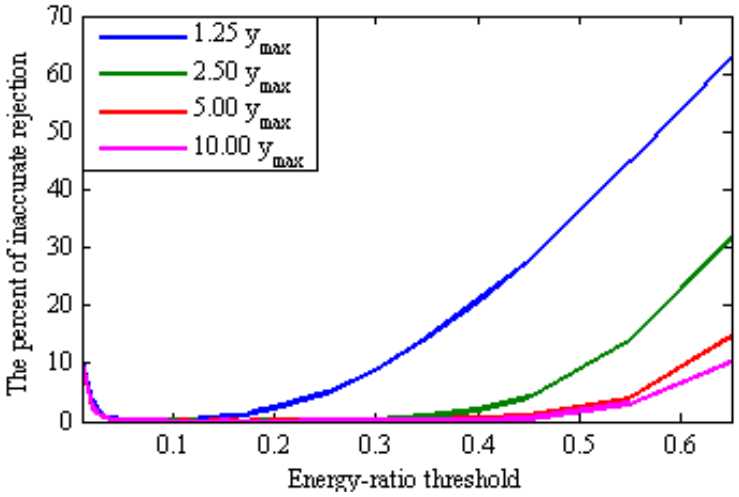


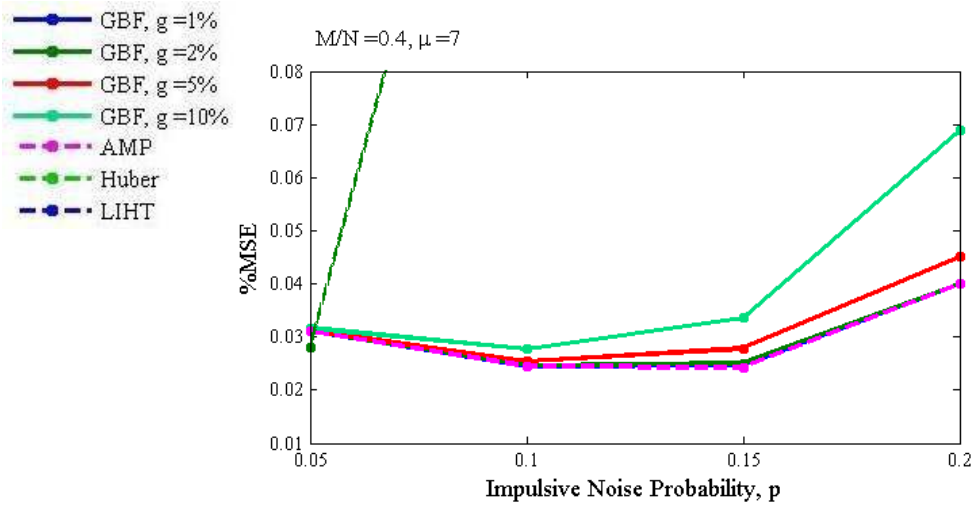
Figure 4.2: The relationship between the energy ratio threshold (η) and the percent of inaccurate rejection. Different colors depict different magnitudes of impulsive noise. y_{max} is the maximum magnitude of the elements in the noiseless **y**.

4.2.2 Performance Evaluation

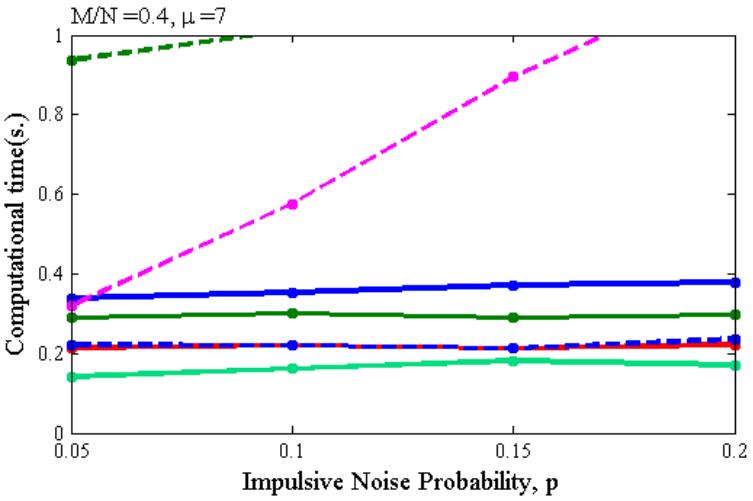
i. First Dataset

Figure 4.3 shows the performances of GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) for the first dataset. Different colors of the bold line indicate different gap resolutions (g). g was set to 1%, 2%, 5% and 10%. Figure 4.3(a) indicated that g of 1%, 2% and 5% provided the comparable %*MSE* to AMP, while HUBER and LIHT provided very high error in the reconstruction. At $p = 0.2$ and $g = 10\%$, the %*MSE* sharply increased, so $g = 10\%$ was not robust in the high noise environment. In most cases, when g was higher (lower resolution), %*MSE* became higher. At $p = 0.15$, $g = 5\%$ provided lower % *MSE* than $g < 5\%$, because the reconstruction from the noiseless **y** had the *ER* larger than 0.1; consequently, the gap's boundary was set incorrectly.

The computational time of GBF was higher than AMP at $p = 5\%$, $g < 10\%$ (Figure 4.3 (b)). However, at $p \geq 10\%$, the computational time of GBF was lower. The result was in accordance with Table 3.1 (Section 3.3), i.e. the computational time of AMP increased with p; whereas, the one of GBF was almost the same. The computational time of GBF was lower when g was higher, since the number of



(a)



(b)

Figure 4.3: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) in the first dataset. (a) average %*MSE* and (b) average computational time. g is the gap resolution in GBF. In this experiment M/N was set at 0.4 and μ of the impulsive noise is $7y_{max}$.

iteration in GBF was inversely varied with *g*. Though the computational time of LIHT was low, the reconstruction error was too high to be considered as an efficient algorithm. HUBER had the highest computational time in all cases.

ii. Second Dataset

Evaluation by PSNR and Computational time

Figures 4.4 - 4.6 shows the performance comparison between GBF (bold line) and the three control algorithms (dashed lines) at different μ when M/N = 0.3. Different colors of the bold lines indicate different g's. Five values of g (1%, 2%, 5%, 10% and 20%) were tested. Among the dashed lines, the magenta, blue and green lines indicate the performance of AMP, HUBER and LIHT, respectively. In all nine images, Figures (a) and (b) shows the PSNR and computational time at different p, respectively.

The performances of AMP and GBF in term of PSNR were comparable in all p and μ; at $p \leq 0.15$ the performance of HUBER was comparable to the lowest performance of GBF (Figures 4.4 – 4.6). PSNR of the reconstruction by HUBER drastically dropped when p = 0.2 at $\mu = 5y_{max}$ and $7y_{max}$. Furthermore, HUBER has visibly lower PSNR at $\mu = 10y_{max}$. LIHT provided the lowest PSNR in all cases. By further investigating the performance of GBF at different g, it was founded that g = 1%, 2% and 5% provided comparable performance to AMP (less than 1 dB PSNR difference), while g = 10% and 20% had notably lower PSNR (more than 3 dB PSNR difference). This is due to the removal of more noiseless elements in reconstruction when g was larger. The PSNR difference between AMP and GBF was less distinct when μ was higher.

Regarding the computational time at M/N = 0.3, AMP, GBF and LIHT had the comparable computational time at p = 0.05; however, the computational time of AMP increased linearly to p. The computational time of GBF and LIHT was about the same for all p. By further investigating the computational time of GBF at different g, it was founded that g = 1%, 2% and 5% provided comparable performance to LIHT. HUBER had the highest computational time in most case. This was to be expected since HUBER was the only technique that was not greedy. Similar to the first dataset, the computational time of GBF was lower when g was higher.

The PSNR and computational time at M/N = 0.35 and 0.4 followed the same trend as M/N = 0.3. The results at M/N = 0.35 and 0.4 were shown in Section A.1 of Appendix A. When the PSNR of the reconstruction at different M/N but the same p was compared, it revealed that there was the strong influence of M/N to the efficiency of LIHT; higher M/N led to higher PSNR.

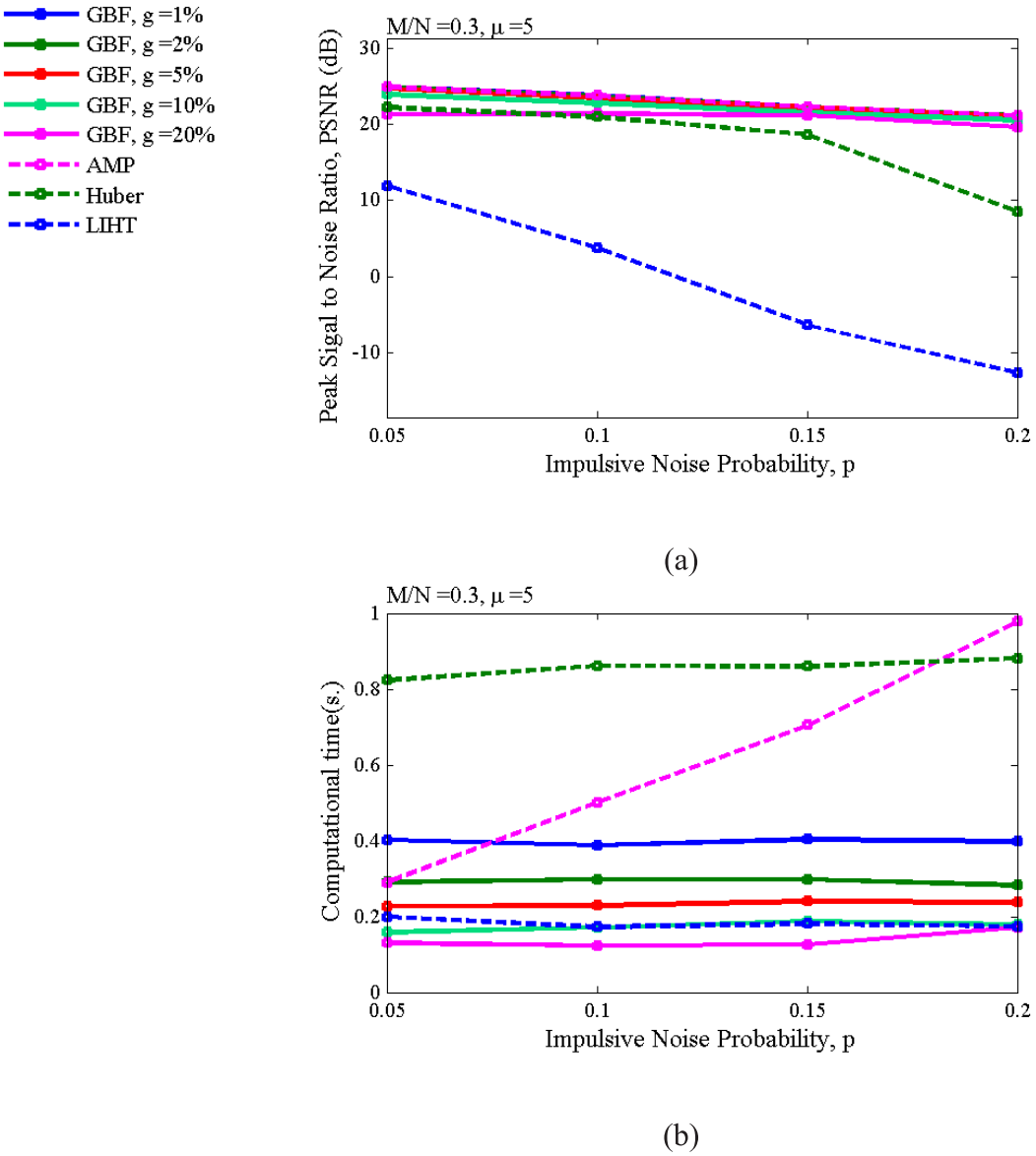


Figure 4.4: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.30, and μ=5y_{max}. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

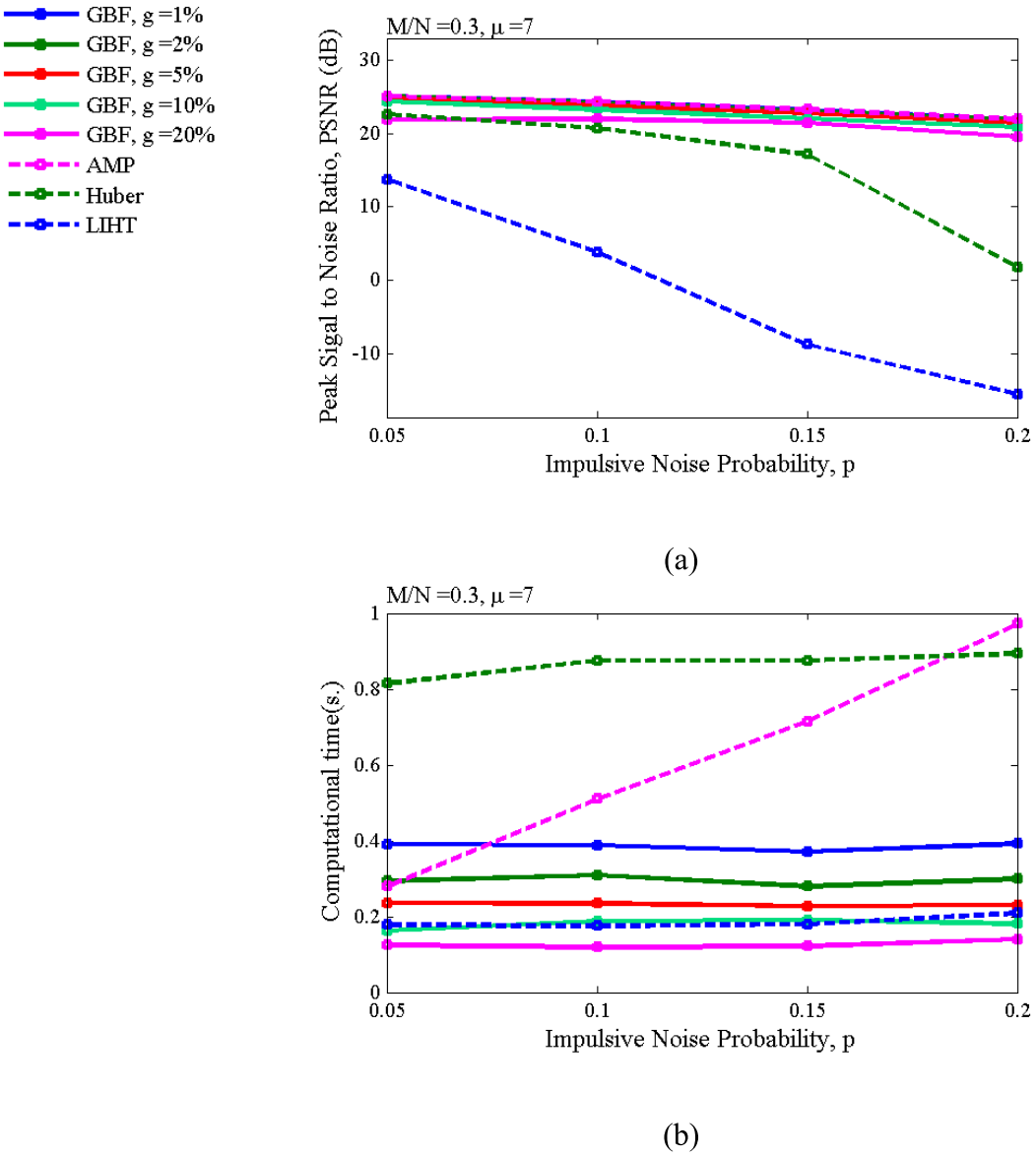


Figure 4.5: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.30, and $\mu=7y_{max}$. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

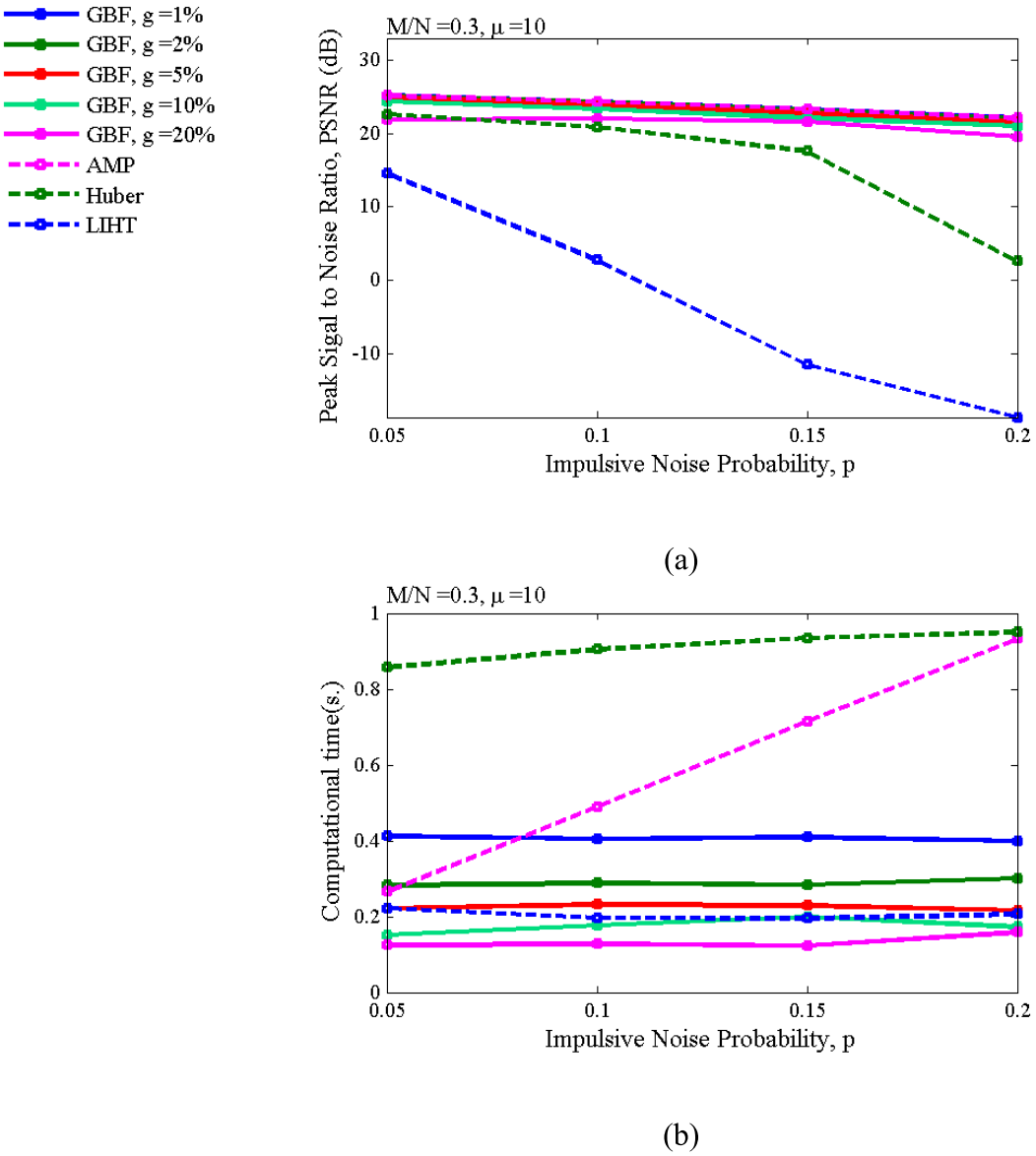


Figure 4.6: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.30, and μ=10y_{max}. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

From Figures 4.4 - 4.6 and the additional results from Section A.1 in Appendix A (Figures A.1 - A.6), the following conclusion could be drawn.

- LIHT was equipped with Lorentzian penalty function which was suitable for impulsive noise rejection and had fast computational time; however, it provided the lowest PSNR because the iterative hard thresholding in LIHT required large M/N (>0.4). Since the benefit of CS was small M/N, LIHT was not efficient for normal images.
- HUBER provided good reconstruction result at $p < 0.20$ and $\mu < 10y_{max}$; however, it required approximately twice the computational time than the other techniques. The lower performance at higher p and μ was expected since the error from impulsive noise was not bounded in Huber penalty function.
- AMP provided the highest PSNR among AMP, HUBER and LIHT; however, it was not suitable for the environment where p was high because its computational time became very high.
- GBF with g = 1%, 2% and 5% provided the comparable PSNR to AMP, while the PSNR of GBF with g = 10% and 20% notably decreased. GBF with g = 5% provided comparable or lower computational time than AMP, thus g = 5% was optimal.
- Except at $p < 0.05$, GBF provided the comparable PSNR, while its computational time was less than AMP's. GBF should be used in place of AMP when $p \geq 0.05$.

Evaluation by Visual Inspection

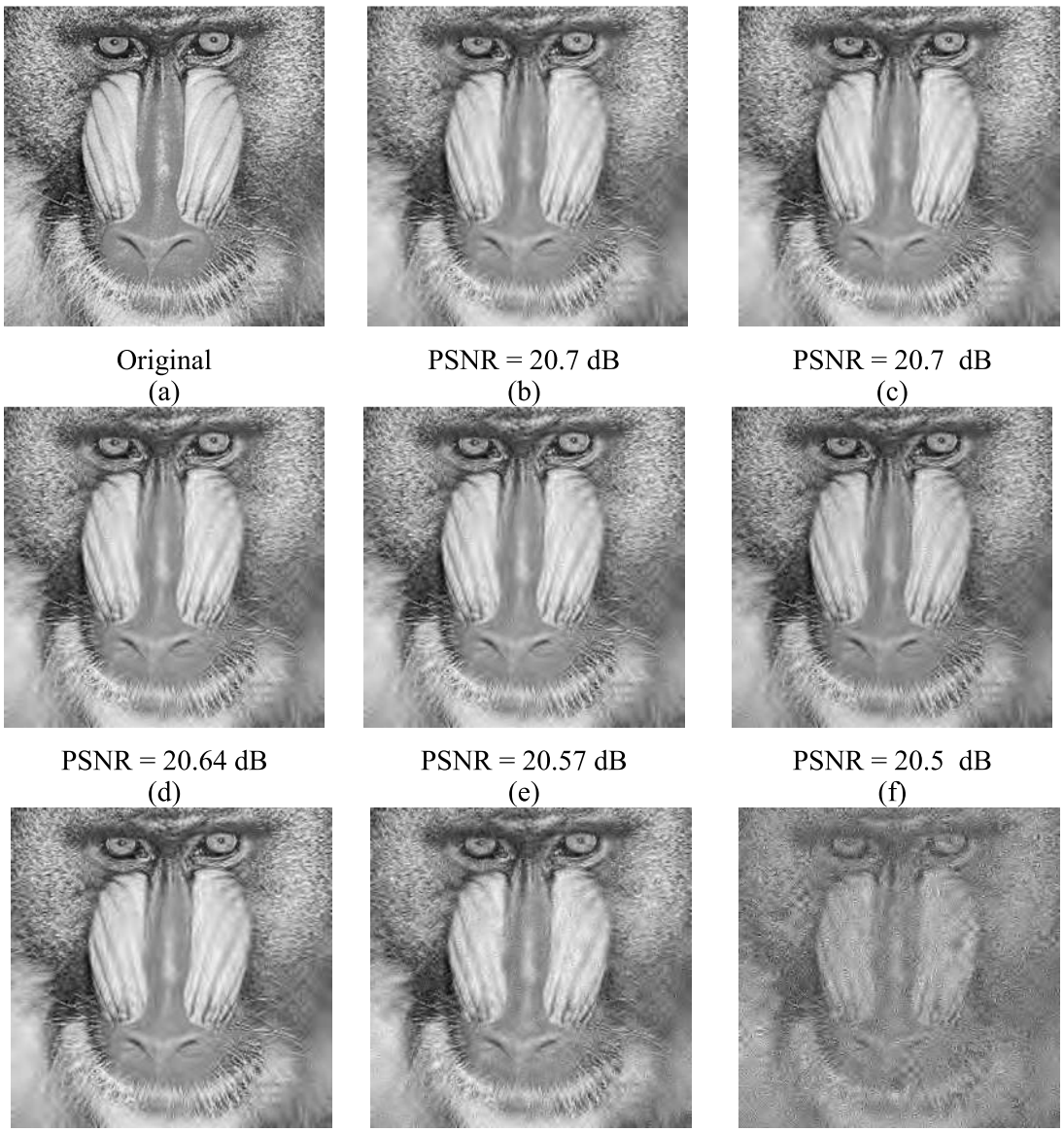
When $\mathbf{y}$ was corrupted by impulsive noise at $p = 0.1$, $M/N = 0.40$ and $\mu = 10y_{max}$, Figures 4.7 - 4.11 show the reconstructed Pepper, Mandrill, Lena, Ripple and Circle, respectively, Figures (a) in all three images show the original image; Figures (b), (c), (d), (e) and (f) show the images reconstructed by OMP-PKS when the preprocessing was GBF at g = 1%, 2%, 5%, 10% and 20%, respectively; Figure (g) show the image reconstructed by OMP-PKS when the preprocessing was AMP; Figures (h) and (i) show the image reconstructed by HUBER and LIHT, respectively.

Original (a) | PSNR = 27.86 dB (b) | PSNR = 27.86 dB (c)

PSNR = 27.8 dB (d) | PSNR = 27.7 dB (e) | PSNR = 27.56 dB (f)

PSNR = 27.86 dB (g) | PSNR = 24.8 dB (h) | PSNR = 15.96 dB (i)

Figure 4.7: The reconstruction of Peppers at M/N = 0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GBF at g = 1%, (c) GBF at g = 2%, (d) GBF at g = 5% and (e) GBF at g = 10%, (f) GBF at g = 20% , (g) AMP, (h) HUBER, and (i) LIHT.



Original (a) | PSNR = 20.7 dB (b) | PSNR = 20.7 dB (c)

PSNR = 20.64 dB (d) | PSNR = 20.57 dB (e) | PSNR = 20.5 dB (f)

PSNR = 20.7 dB (g) | PSNR = 19.29 dB (h) | PSNR = 14.97 dB (i)

Figure 4.8: The reconstruction of Mandrill at M/N = 0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GBF at g = 1%, (c) GBF at g = 2%, (d) GBF at g = 5% and (e) GBF at g = 10%, (f) GBF at g = 20% , (g) AMP, (h) HUBER, and (i) LIHT.

Original
(a)

PSNR = 28.24 dB
(b)

PSNR = 28.24 dB
(c)

PSNR = 28.13 dB
(d)

PSNR = 28.11 dB
(e)

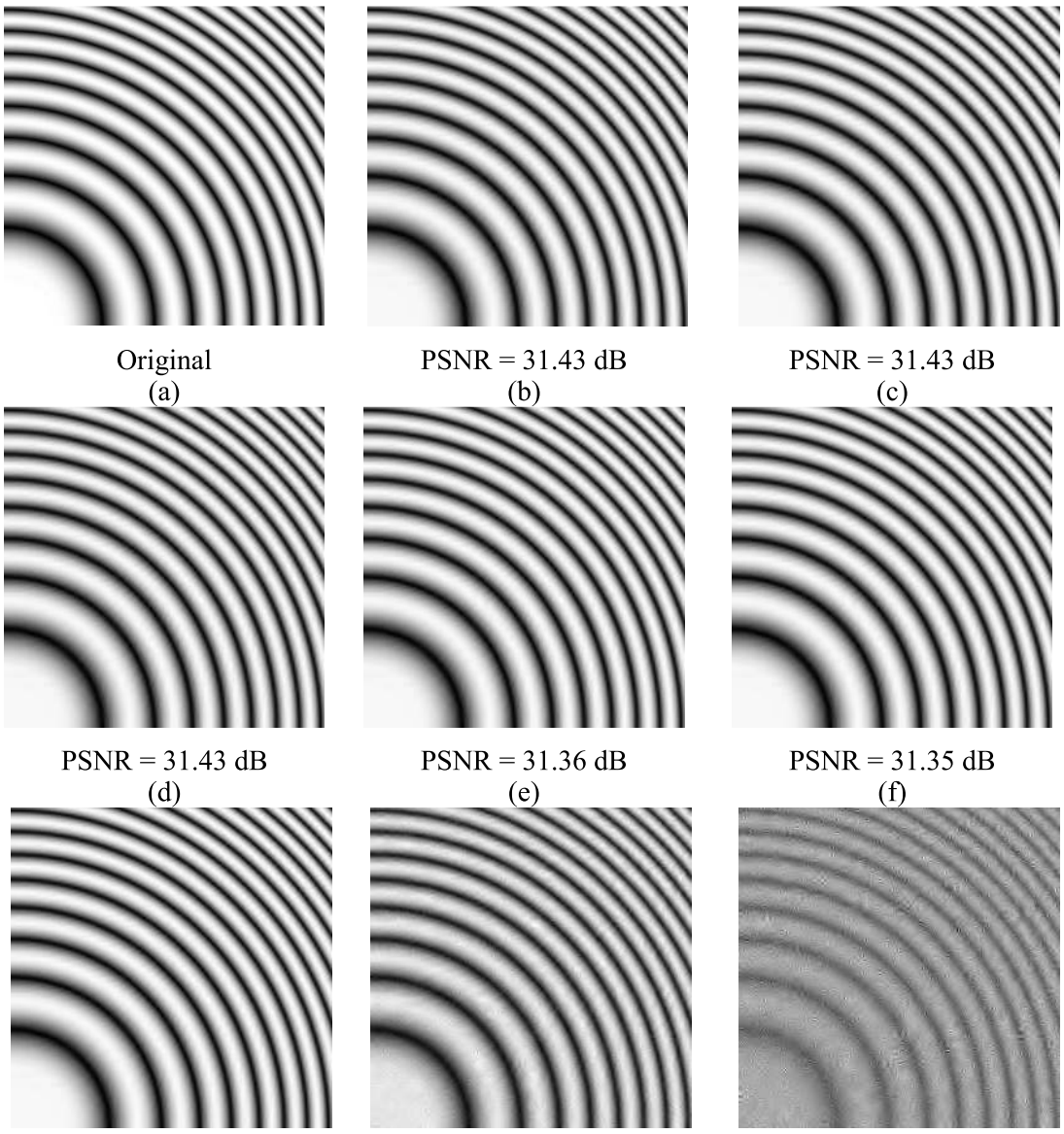
PSNR = 28.06 dB
(f)

PSNR = 28.24 dB
(g)

PSNR = 25.1 dB
(h)

PSNR = 16.19 dB
(i)

Figure 4.9: The reconstruction of Lena at $M/N = 0.40$, $\mu = 7y_{max}$, $p = 0.10$. (a) original image, the reconstructed images by (b) GBF at $g = 1\%$, (c) GBF at $g = 2\%$, (d) GBF at $g = 5\%$ and (e) GBF at $g = 10\%$, (f) GBF at $g = 20\%$, (g) AMP, (h) HUBER, and (i) LIHT.

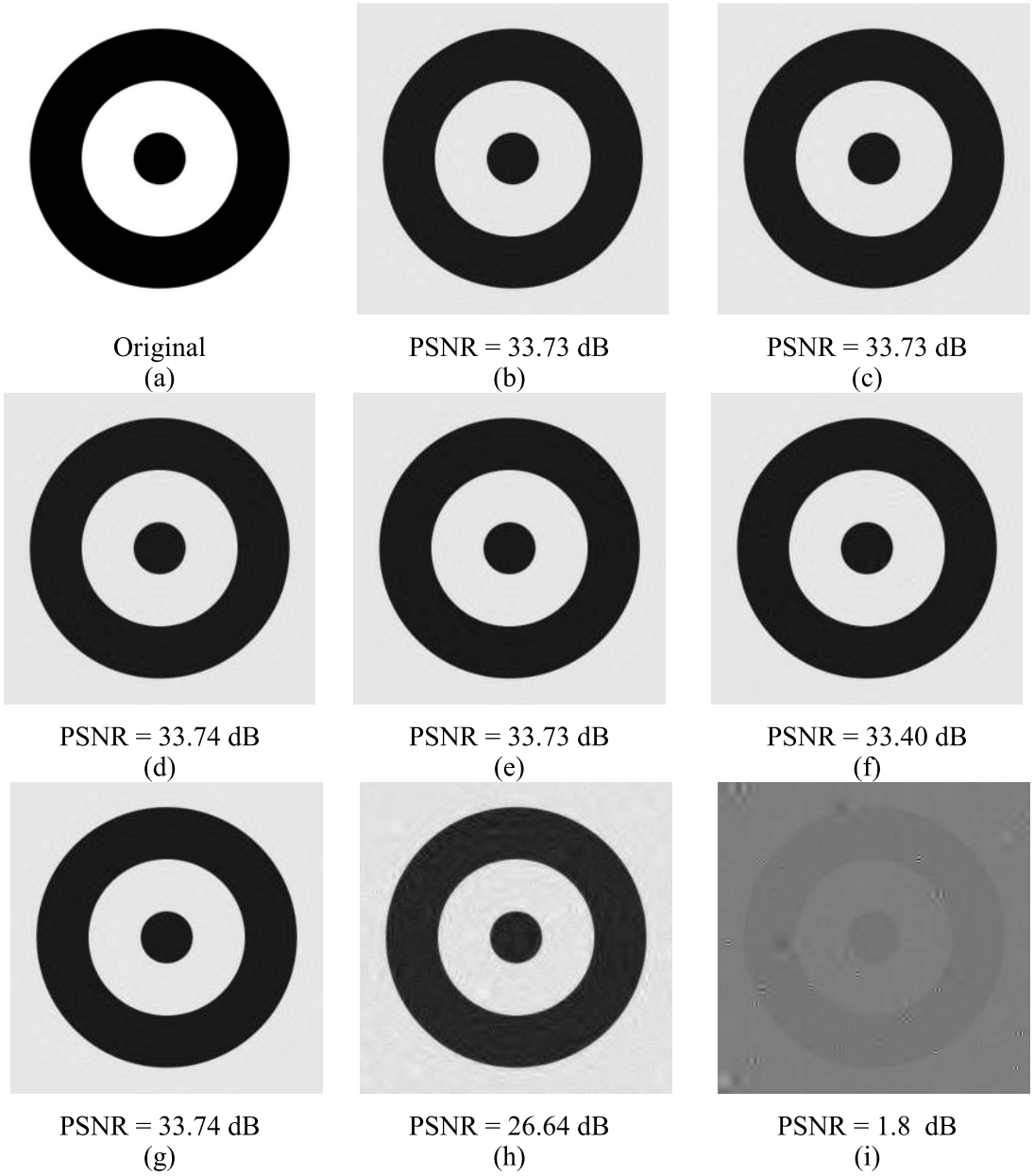


Original (a) | PSNR = 31.43 dB (b) | PSNR = 31.43 dB (c)

PSNR = 31.43 dB (d) | PSNR = 31.36 dB (e) | PSNR = 31.35 dB (f)

PSNR = 31.43 dB (g) | PSNR = 25.59 dB (h) | PSNR = 13.52 dB (i)

Figure 4.10: The reconstruction of Ripples at M/N = 0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GBF at g = 1%, (c) GBF at g = 2%, (d) GBF at g = 5% and (e) GBF at g = 10%, (f) GBF at g = 20% , (g) AMP, (h) HUBER, and (i) LIHT.



Original (a) | PSNR = 33.73 dB (b) | PSNR = 33.73 dB (c)

PSNR = 33.74 dB (d) | PSNR = 33.73 dB (e) | PSNR = 33.40 dB (f)

PSNR = 33.74 dB (g) | PSNR = 26.64 dB (h) | PSNR = 1.8 dB (i)

Figure 4.11: The reconstruction of Circles at M/N = 0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GBF at g = 1%, (c) GBF at g = 2%, (d) GBF at g = 5% and (e) GBF at g = 10%, (f) GBF at g = 20% , (g) AMP, (h) HUBER, and (i) LIHT.

In all cases, there was no distinct visual difference among the reconstructed results of GBFs at g = 1%, 2% and 5% and AMP. The degradation of the reconstructed results was mostly due to the shrinkage thresholding. It can be concluded that g should be set to 5%, since it produced the comparable performance to AMP, while it used the lowest computational time. The figure also indicated that

the reconstruction by HUBER provided slightly worse visual quality, while the one by LIHT was the worst in term of the visual quality.

Because all blocks in Figure 4.7 - 4.11 did not have the same M/N and sparsity rates, the smoothness of all reconstructed images indicated that one threshold of GSSF and AMP were effective for multiple measurement and sparsity rates.

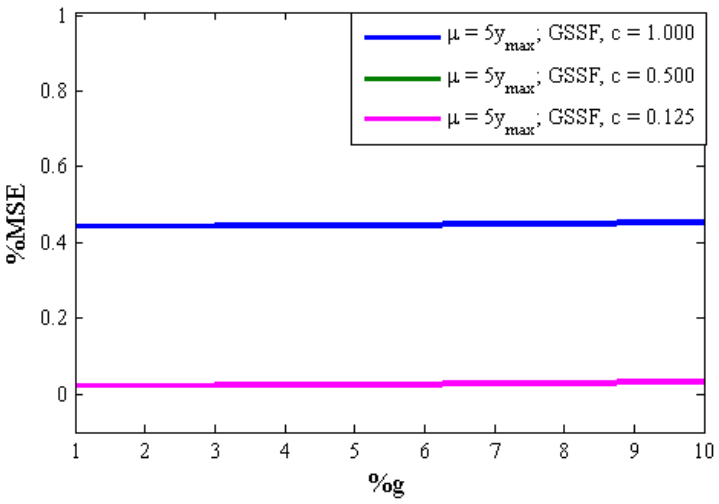
4.3 The Evaluation of GSSF

The evaluation was divided into two main sections: (1) evaluation for the optimal threshold in Section 4.3.1 and (2) the performance evaluation in Section 4.3.2. The first dataset was used in both sections, while the second dataset was used only in Section 4.3.2.

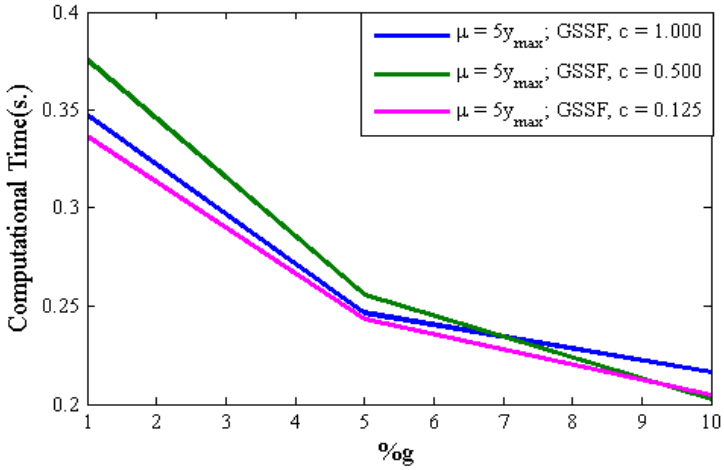
Similar to the evaluation of GBF (Section 4.2), the magnitude of the impulsive noise was set relative to y_{max}. The distribution of the noise magnitude was Gaussian. The deviation was set to y_{max}. The first dataset was corrupted by the noise with $\mu = 5y_{max}$. In the second dataset, three levels of μ (5 y_{max}, 7y_{max} and 10y_{max}) were tested. In both datasets, the experiment was performed on four levels of p (0.05, 0.10, 0.15 and 0.20).

4.3.1 Evaluation for Optimal Threshold

Figure 4.12 shows the performance of GSSF at different g. Different colors depicted different slope scale (c in Equation (3.1)). Figure 4.12(a) indicated that g has little effect to %MSE, while c = 0.5 and 0.125 provided almost the same PSNR. Regarding the computational time (Figure 4.12(b)), the higher g led to the lower computational time, while c had little effect. From the experiment, it can be concluded that g should be set to 10%, since it produced the comparable %MSE to other lower g but with the lowest computational time. The optimal value of c could not be definitely concluded; however, Figure 4.12 indicated that c = 1 should not be used.



(a)



(b)

Figure 4.12: The performance of GSSF at different *g* in term of (a) %MSE and (b) computational time. Different color depicts different *c*.

4.3.2 Performance Evaluation

i. First Dataset

Figure 4.13 shows the performance of GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) in term of %*MSE* (Figure 4.13(a)) and computational time (Figure 4.13(b)). In GSSF, g = 10% had comparable performance and provided the lowest %*MSE* in all cases. Since g was set to 10%, GSF had lower computational time than AMP. Though HUBER was better than LIHT, its reconstruction had more %*MSE* than AMP and GSSF, and consumed more computational time (approximately two times of the one of GSSF).

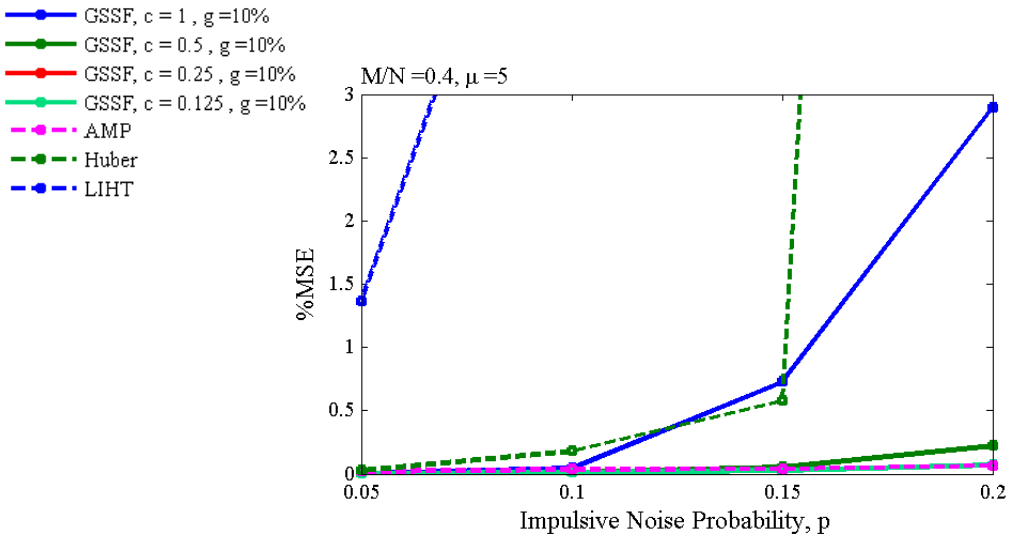
Even though μ in this experiment was different than Section 4.3.1, there was no distinct difference in the performance when c = 0.5, 0.25 and 0.125. Thus, the optimal value of c could not be established.

ii. Second Dataset

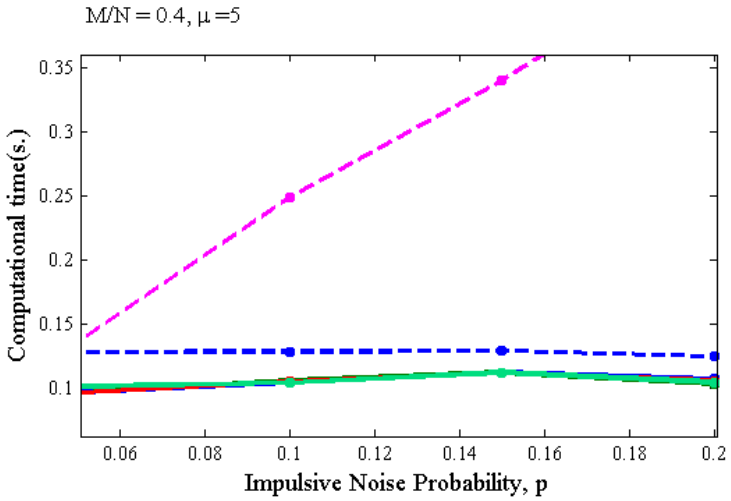
Evaluation by PSNR and Computational time

Figure 4.14 - 4.16 show the PSNR and computational time at different μ when M/N = 0.3. In most cases, AMP (dashed magenta line) provided the highest PSNR; however, the PSNR from GSSF (bold line) was only slightly lower than AMP. In term of the computational time, GSSF consumed less computational time than AMP in most cases, since g = 10% was used. However, the computational time in GSSF could be changed in both increasing and decreasing manners when p increased. In most cases, HUBER (dashed green line) provided less PSNR than AMP and GSSF, while its computational time was two times of GSSF. Though LIHT (dashed blue line) was among the fastest technique, it provided much lower PSNR.

The PSNR and computational time at M/N = 0.35 and 0.4 followed the same trend as M/N = 0.3. The results at M/N = 0.35 and 0.4 were shown in Section A.2 of Appendix A. It was found that the PSNR different became less when M/N was higher.



(a)



(b)

Figure 4.13: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) in the first dataset (a) average %*MSE* and (b) average computational time. g and c are the gap resolution and slope scale in GSSF, respectively. In this experiment M/N was set at 0.4 and μ of the impusive noise was set to $5y_{max}$.

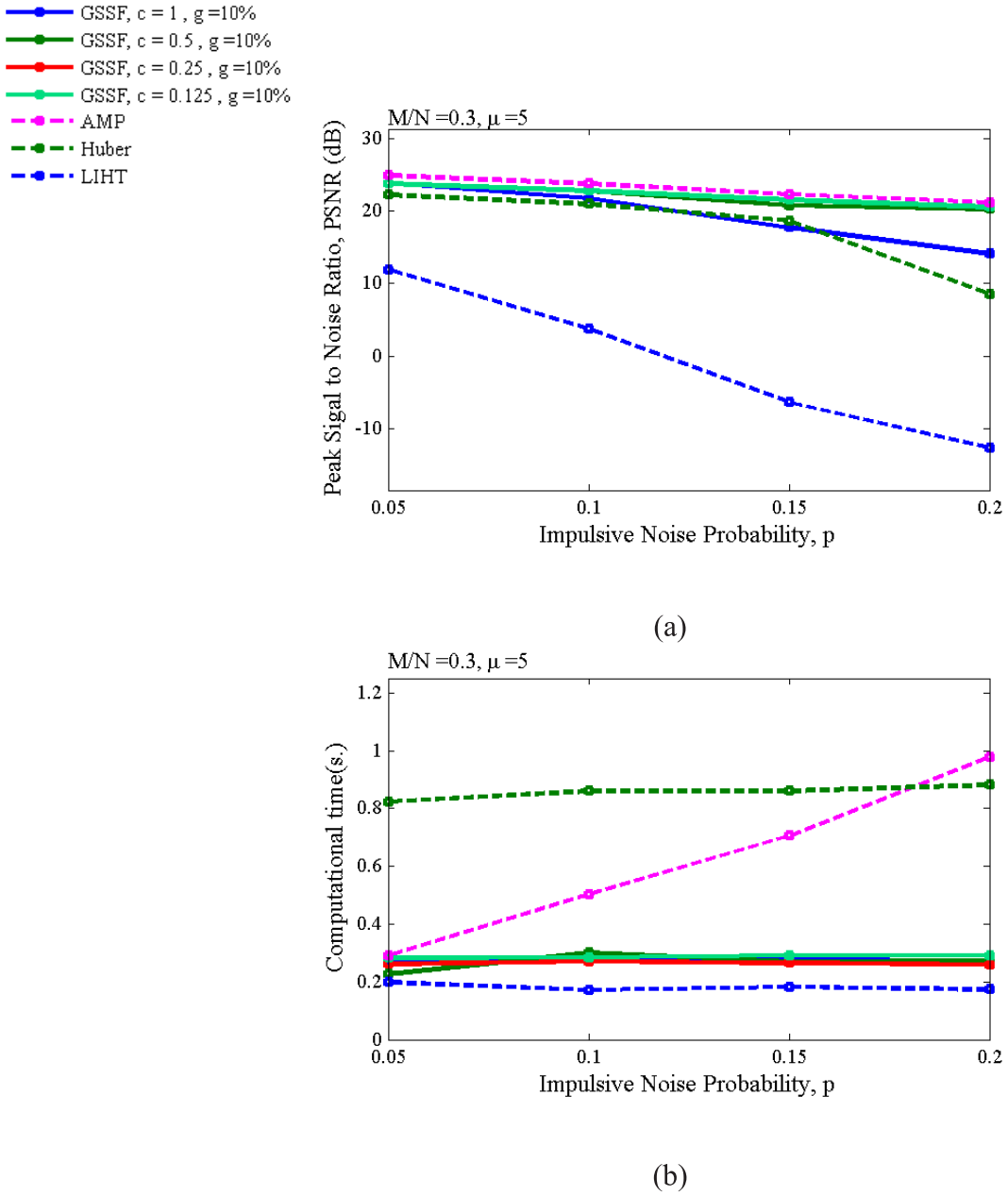


Figure 4.14: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.30, and $\mu = 5y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

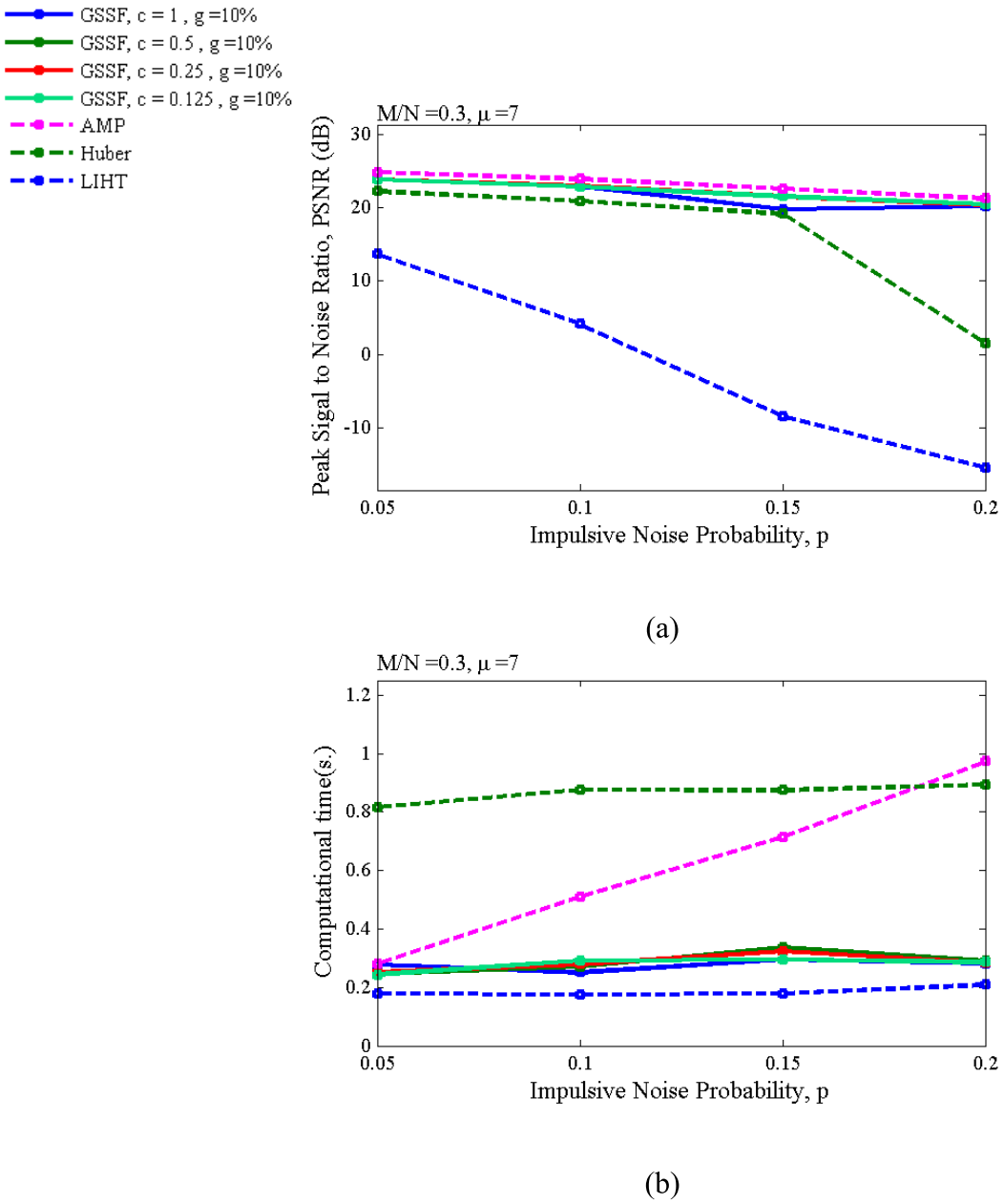


Figure 4.15: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.30, and $\mu = 7y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

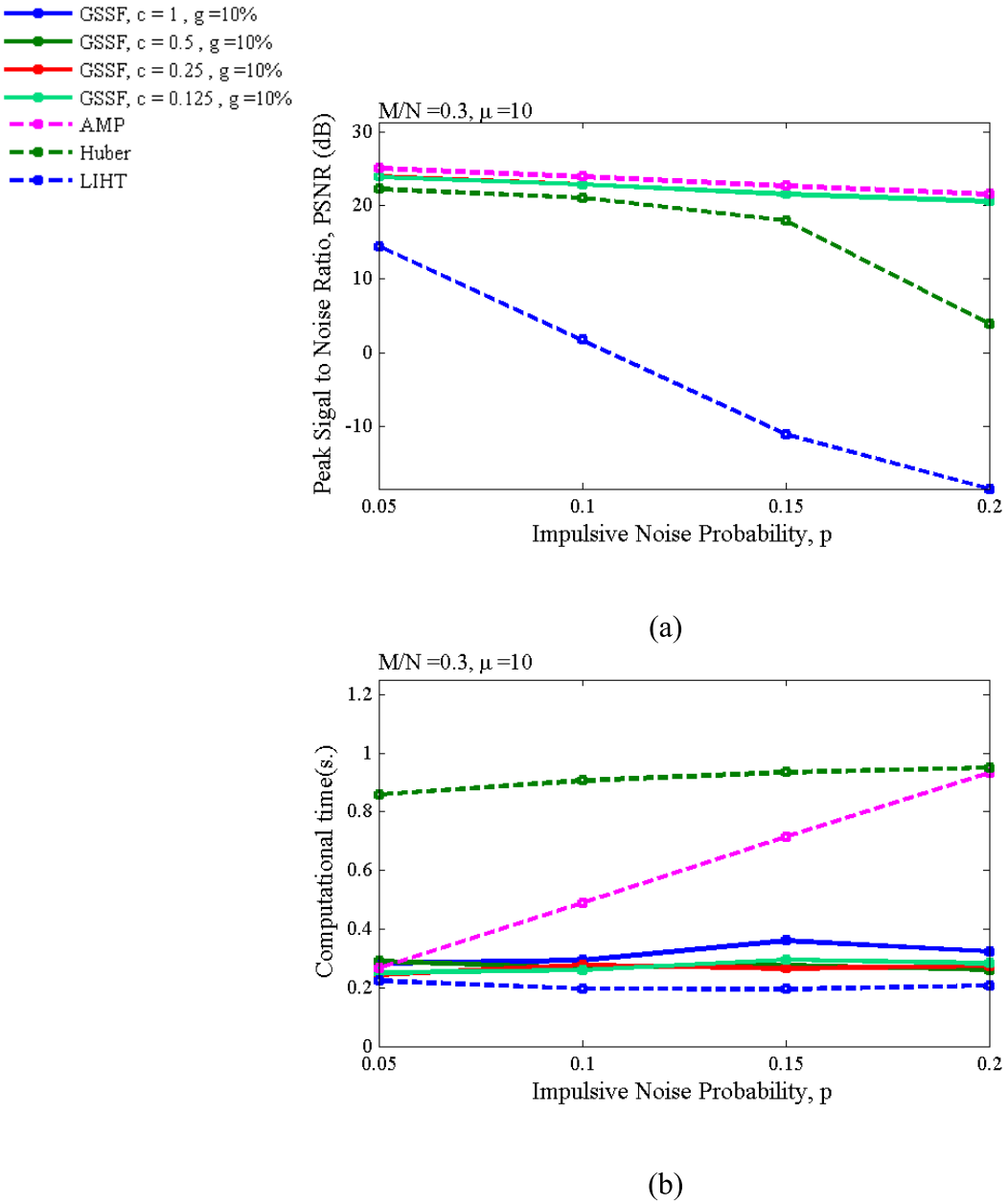


Figure 4.16: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.30, and $\mu = 10y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

Even though three different M/N's and three different μ were used in the experiment, there was no distinct different when c was varied. Though no optimal value of c could be established, the lack of performance drop was not undesired. It indicated that it was unnecessary to experiment for the exact value of c.

From Figure 4.14 - 4.16 and the additional results from Section A.2 in Appendix A (Figures A.7 - A.12), the following conclusion could be drawn.

- GSSF provided comparable PSNR to AMP but required less computational time. The experiment showed that the computational time of GSSF was less in all cases when $p \geq 0.10$. Similar to GBF, GSSF provided higher PSNR than HUBER and LIHT in all cases.
- GSSF did not require the carefully selected c. The performance at c = 0.125, 0.25 and 0.5 were almost the same; however, c should not be too large. The large value of c would lead to more difficulty in accepting the existence of impulsive noise. The experiment indicated that the risk of accepting the impulsive noise outweighed the risk of rejecting the noiseless elements.
- GSSF should be used in place of AMP when $p \geq 0.05$.

Evaluation by Visual Inspection

Figures 4.17 - 4.21 show the reconstructed Peppers, Mandrills, Lena, Ripples and Circles, respectively. $\mathbf{y}$ was corrupted by impulsive noise at $p = 0.1$ and $\mu = 7y_{max}$. M/N was set to 0.4. Figure (a) shows the original image. When $c \neq 1$, the reconstruction results of AMP (Figure (f)) and GSSF (Figures (b) - (d)) were almost the same. Some visual degradation was found, when c of GSSF was set to 1 (Figure 4.15(e)). The reconstruction by HUBER (Figures (g)) was visually worse than the ones of AMP and GSSF, while the reconstruction of LIHT (Figure (h)) were far worse than the ones of AMP and GSSF.

Because all blocks in the image did not have the same measurement rate and the sparsity level, the smoothness of all reconstructed images indicated that one threshold of GSSF and AMP were effective for multiple measurement rates and sparsity levels.

Original
(a)

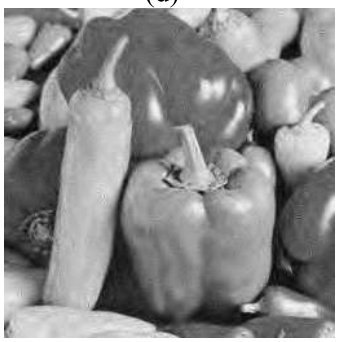
PSNR = 27.8 dB
(b)

PSNR = 27.8 dB
(c)

PSNR = 27.8 dB
(d)

PSNR = 27.8 dB
(e)

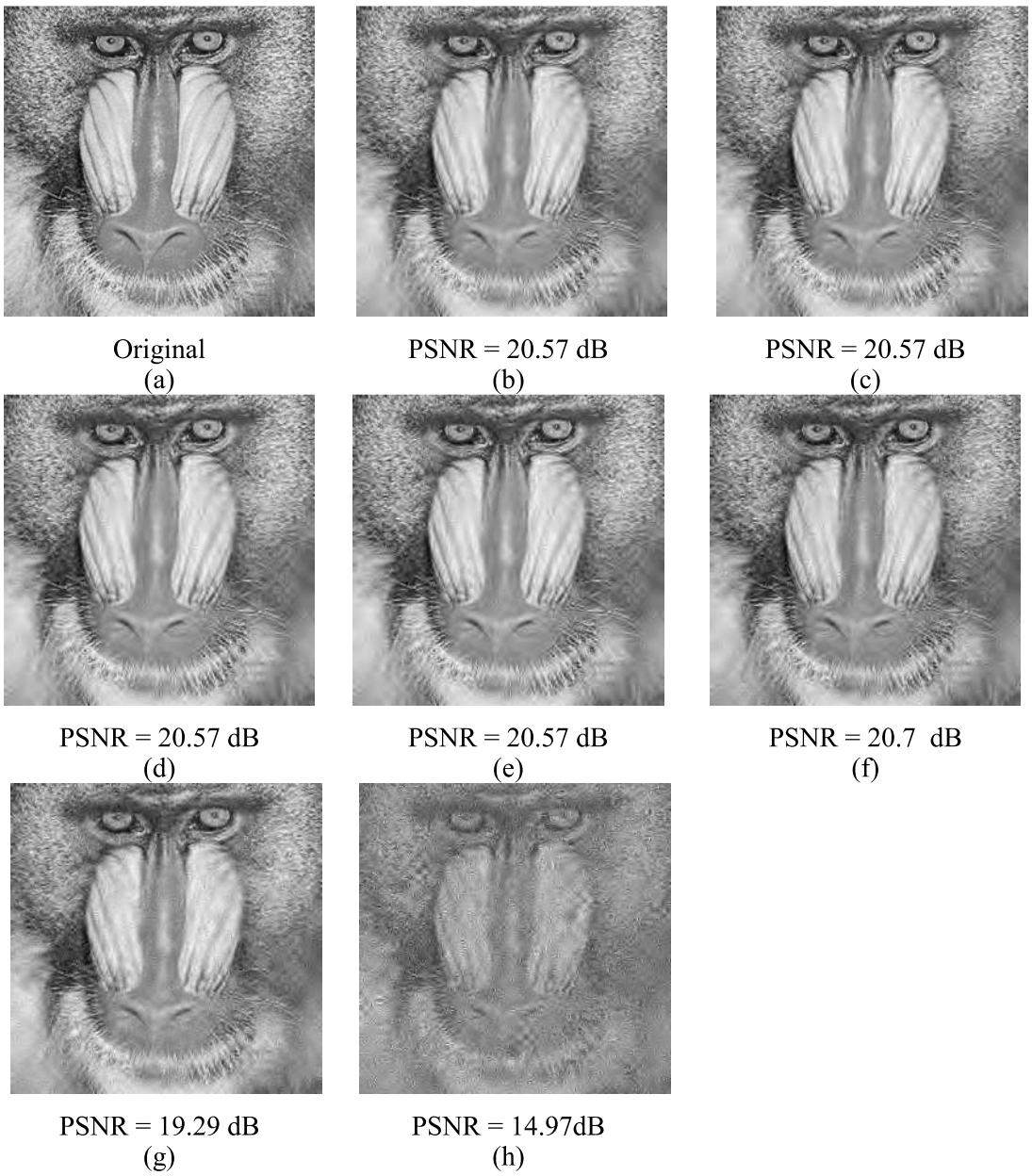
PSNR = 27.86 dB
(f)



PSNR = 24.8 dB
(g)

PSNR = 15.96 dB
(h)

Figure 4.17: The reconstruction of Peppers at M/N=0.40, μ =7y_{max}, p = 0.10. (a) original image, the reconstructed images by (b) GSSF at c = 0.125, (c) GSSF at c = 0. 25, (d) GSSF at c = 0.5 , and (e) GSSF at c = 1, (f) AMP, (g) HUBER and (i) LIHT.



Original
(a)

PSNR = 20.57 dB
(b)

PSNR = 20.57 dB
(c)

PSNR = 20.57 dB
(d)

PSNR = 20.57 dB
(e)

PSNR = 20.7 dB
(f)

PSNR = 19.29 dB
(g)

PSNR = 14.97dB
(h)

Figure 4.18: The reconstruction of Mandrill at M/N=0.40, μ =7y_{max}, p = 0.10. (a) original image, the reconstructed images by (b) GSSF at c = 0.125, (c) GSSF at c = 0. 25, (d) GSSF at c = 0.5 , and (e) GSSF at c = 1, (f) AMP, (g) HUBER and (i) LIHT.

Original
(a)

PSNR = 28.11 dB
(b)

PSNR = 28.11 dB
(c)

PSNR = 28.1 dB
(d)

PSNR = 28.1 dB
(e)

PSNR = 28.24 dB
(f)

PSNR = 25.1 dB
(g)

PSNR = 16.19 dB
(h)

Figure 4.19: The reconstruction of Lena at M/N=0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GSSF at c = 0.125, (c) GSSF at c = 0. 25, (d) GSSF at c = 0.5 , and (e) GSSF at c = 1, (f) AMP, (g) HUBER and (i) LIHT.

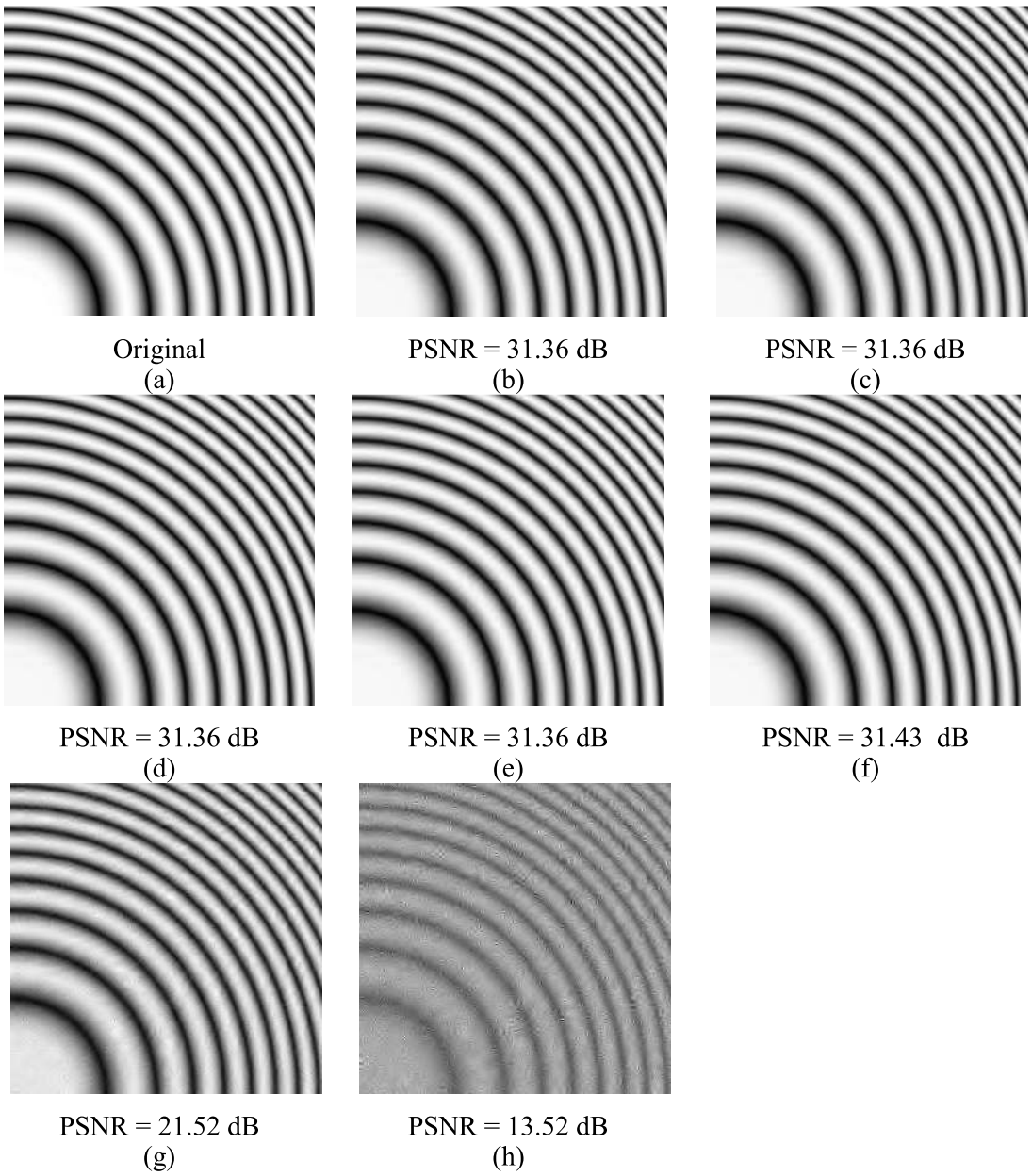
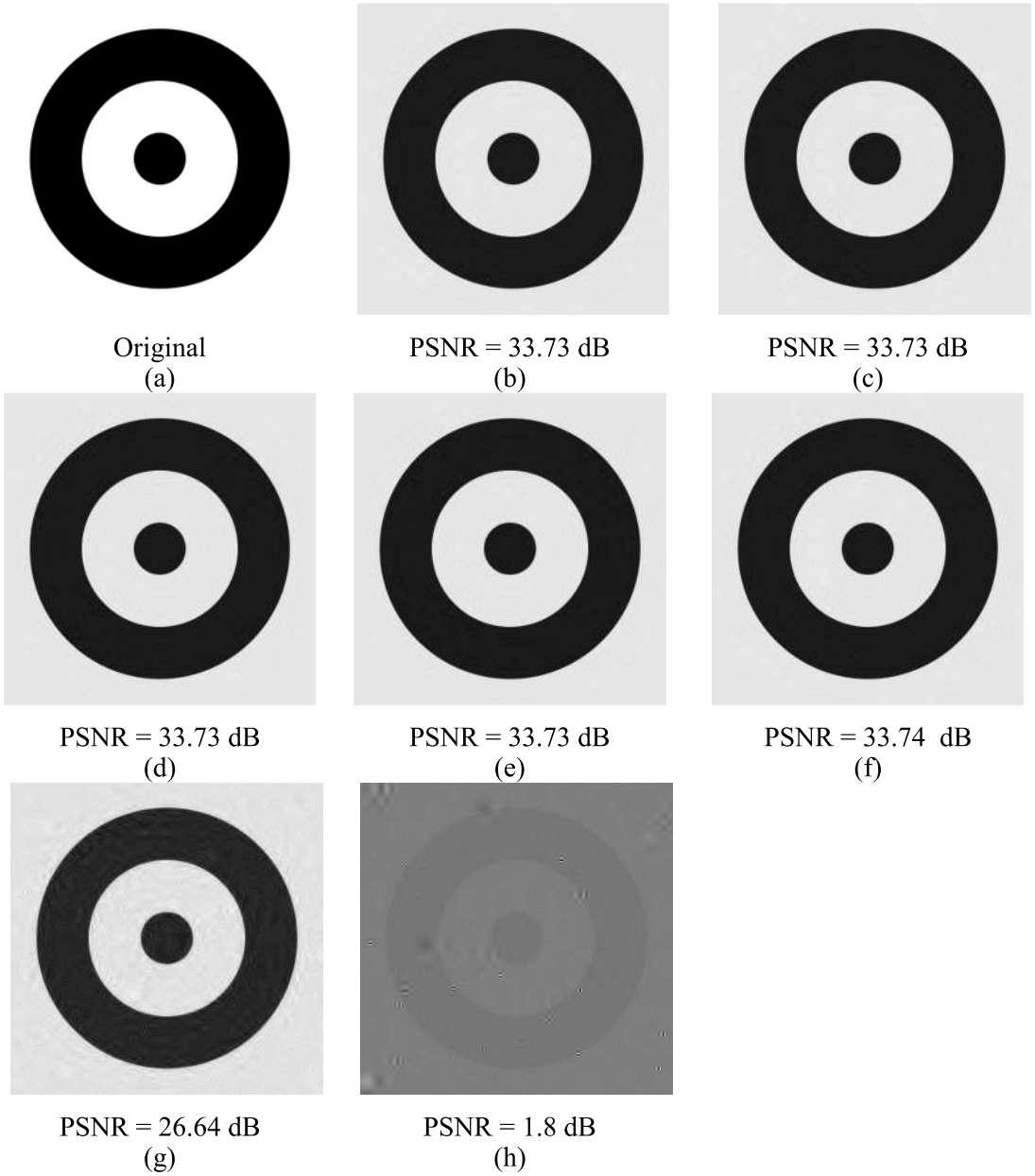


Figure 4.20: The reconstruction of Ripple at M/N=0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GSSF at c = 0.125, (c) GSSF at c = 0. 25, (d) GSSF at c = 0.5 , and (e) GSSF at c = 1, (f) AMP, (g) HUBER and (i) LIHT.



Original
(a)

PSNR = 33.73 dB
(b)

PSNR = 33.73 dB
(c)

PSNR = 33.73 dB
(d)

PSNR = 33.73 dB
(e)

PSNR = 33.74 dB
(f)

PSNR = 26.64 dB
(g)

PSNR = 1.8 dB
(h)

Figure 4.21: The reconstruction of Circle at M/N=0.40, $\mu = 7y_{max}$, p = 0.10. (a) original image, the reconstructed images by (b) GSSF at c = 0.125, (c) GSSF at c = 0. 25, (d) GSSF at c = 0.5 , and (e) GSSF at c = 1, (f) AMP, (g) HUBER and (i) LIHT.

4.4 Comparative Evaluation Between GBF and GSSF

Since the performances of GBF and GSSF were almost the same, GBF and GSSF were evaluated with each other. In this section, the second dataset was corrupted by the noise with the same characteristic as Sections 4.2.2 and 4.3.2, where the distribution of the noise magnitude was Gaussian with the deviation of y_{max}. The experiment was performed on three values of μ ($5y_{max}$, $7y_{max}$ and $10y_{max}$) and four values of p (0.05, 0.10, 0.15 and 0.20).

Figures 4.22 show the PSNR (Figure (a)) and the computational time (Figure (b)) of GBF (bold line) and GSSF (dashed lines) at M/N = 0.30. g of GBF was set to 5%, while c and g of GSSF were set to 0.125 and 10%, respectively. The PSNR of GBF was slightly higher in most cases. At the same p, the higher μ led to higher PSNR, because the change in the *ER* was more distinct.

Even though GBF at g = 5% and GSSF at g = 10% require the same number of reconstruction, the computational time of GBF was lower. GBF directly used *ER*, so there was no additional calculation step as GSSF which required the recalculation of *ER* slope in every reconstruction. Furthermore, the computational time of GBF was more stable. Thus, in an application that imposes severe constraint to computational time, GBF is the more suitable of the two.

The results at M/N = 0.35 and 0.40 followed the same trend as M/N = 0.30 and were provided in Section A.3 of Appendix A (Figures A.13 - A.14). It was also found that the PSNR difference between GBF and GSSF was less when M/N increased.

Figures 4.23 and 4.24 show the reconstruction examples at M/N = 0.35 and 0.40, respectively. The impulsive noise was set at p = 0.20 and μ= $5y_{max}$. There was no distinct difference between the reconstruction by GBF and GSSF. The benefit of using the change in *ER* to detect the impulsive noise in GSSF was not found in the experiment. This was caused by the preference for smaller c in GSSF. The small c led to the threshold that detected the impulsive noise with smaller change in *ER* (less steep slope), i.e. the possibility of removing noiseless elements was higher. Thus, the

benefit of avoiding removing the noiseless elements was negated by the preference of small *c*.

In conclusion, GBF and GSSF provided comparable PSNR but the computational time of GBF was more predictable and usually smaller than GSSF. Thus, GBF is recommended as the better of the two algorithms.

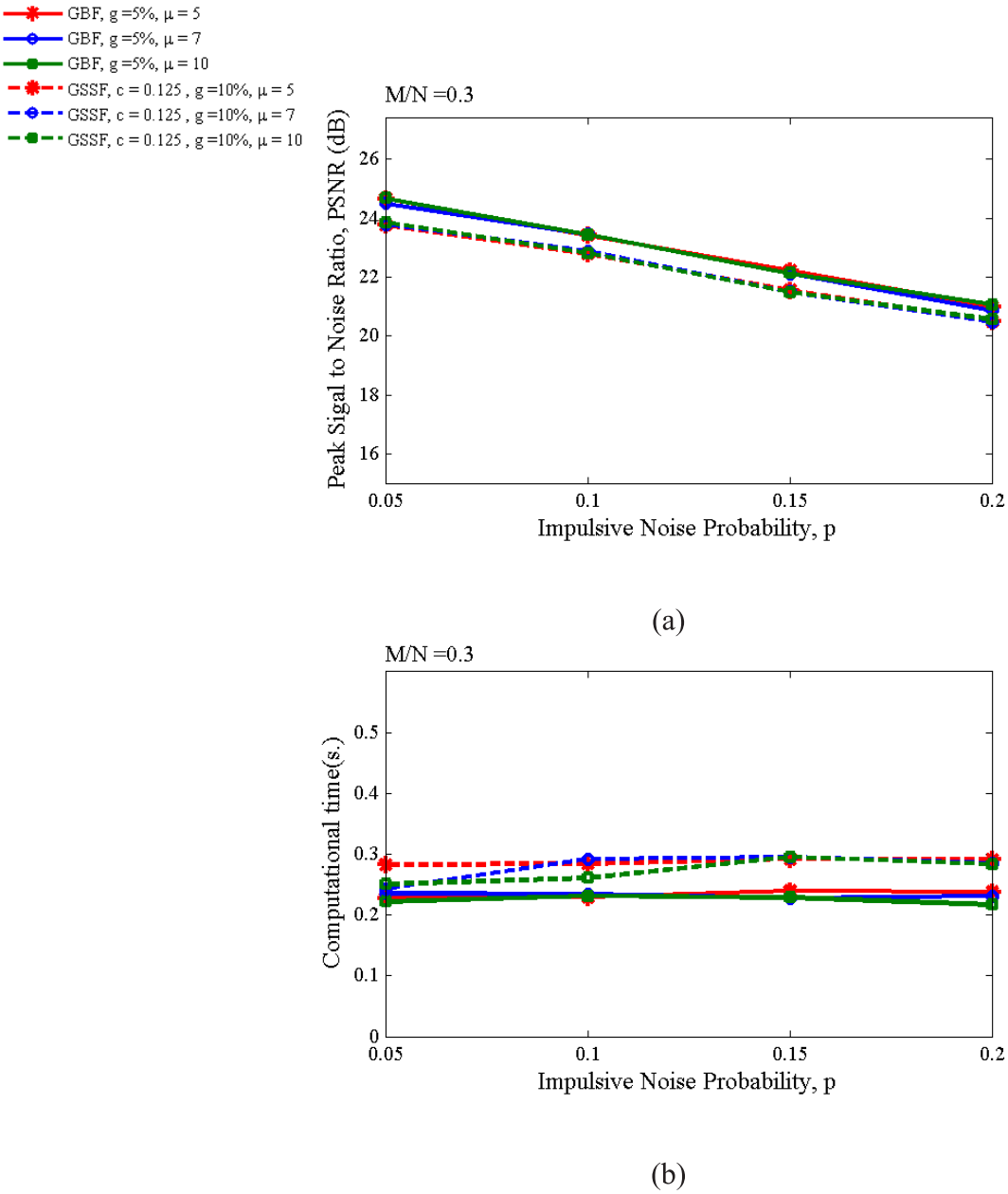


Figure 4.22: Performance comparison between GBF (g = 5%, bold line), and GSSFs (g = 10%, c = 0.125, dashed line) when M/N = 0.30. (a) average PSNR and (b) average computational time (per block).

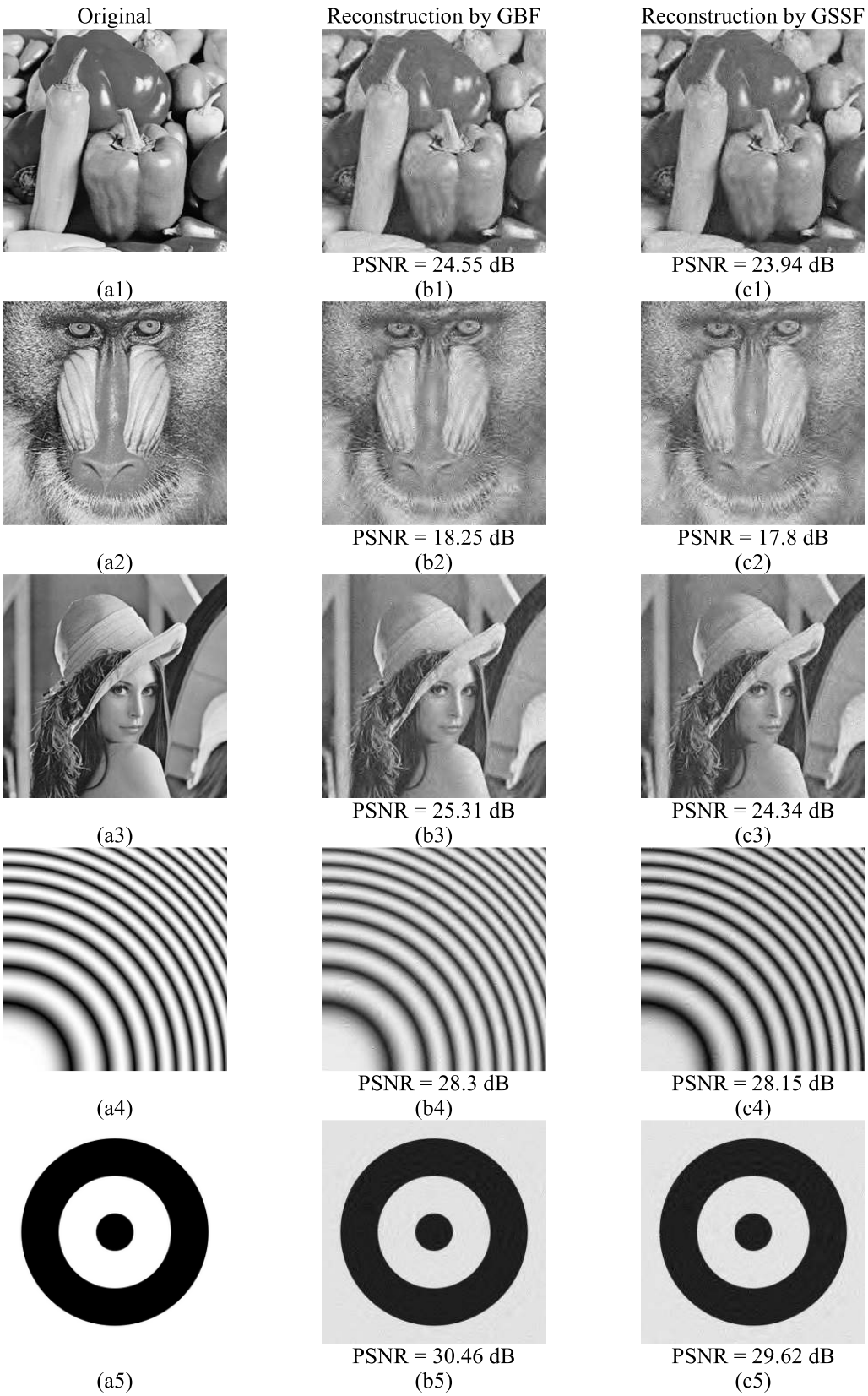


Figure 4.23: The example of the reconstruction by GBF (g = 5%) and GSSF (g = 10% and c = 0.125) at M/N = 0.35, p = 0.2, and $\mu = 5y_{max}$.

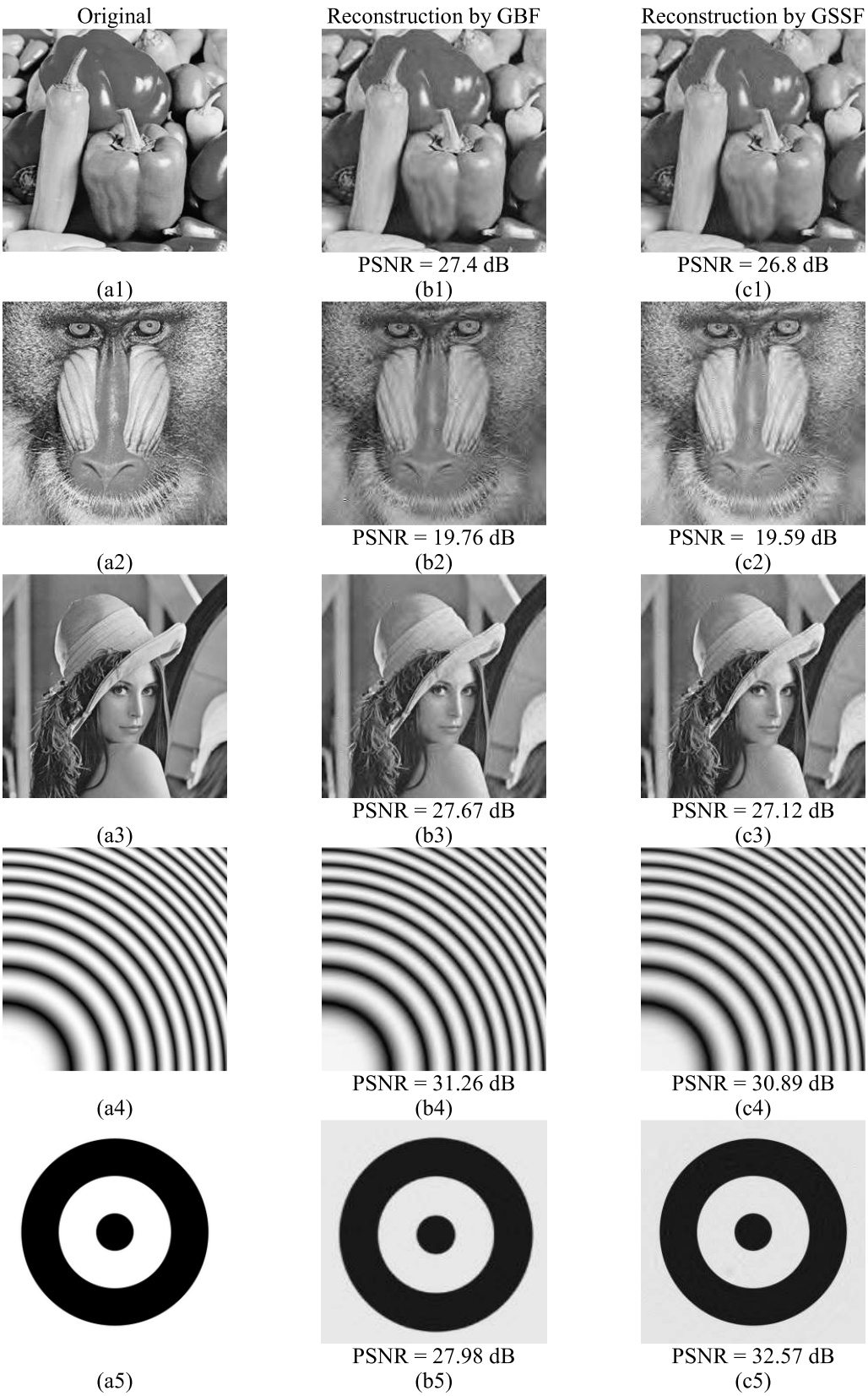


Figure 4.24: The example of the reconstruction by GBF (g = 5%) and GSSF (g = 10% and c = 0.125) at M/N = 0.4, p = 0.2, and $\mu = 5y_{max}$.

4.5 Evaluation for the Limitation of Noise Removal by ER

In the previous three sections, the experiment was performed within the scope of the thesis and there were not any cases that GBF, GSSF as well as AMP failed to reject impulsive noise. In order to investigate for the cases where these three algorithms failed, the experiment in this section was performed with the data not within the scope of the thesis. Since *ER* was used to detect the noise existence, the three algorithms would fail if *ER* was no longer a valid indicator. Therefore, the limitation of *ER* was investigated.

4.5.1 Type of Image

Images used in Sections 4.2 - 4.4 were selected such that they were sparsified by octave tree DWT. Wavelet shrinkage thresholding is possible, when information is sparse in every subband but LL subband. However, the sparsity level is not the same among images. When images have lots of fine details, the wavelet shrinkage thresholding will lead to severe degradation. Figure 4.25 show examples of the failed sparsification by the wavelet shrinkage thresholding. Compared to the original image (the left column), lots of details after the wavelet shrinkage thresholding (the right column) were missing. *ER*'s before and after were investigated. It was founded that the images could be categorized in to two groups: (1) images with *ER* higher than 0.1, which was the threshold used in AMP and GBF, and (2) images whose *ER* drop was large (Ex. 32.2% and 28.6% drop in *ER* for Figures (a.3) and (a.4), respectively). Though the wavelet shrinkage thresholding did not cause severe degradation for the images in the second group, the details were blurred out so the images were not used in the experiment.

ER as the indicator for the presence of noise was possible in all experiments, because wavelet shrinkage thresholding ensured that most energy in an image was inside the *LL* subband (which was inside the L_3 subband). Therefore, if the image cannot be sparsified or details are lost by the wavelet shrinkage thresholding, *ER* will not be a good indicator. Consequently, AMP, GBF and GSF will not be effective.

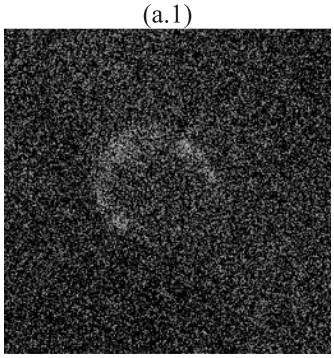
Note that the direct experiment is not possible because the input image (the sparse signal) is degraded; thus, the detection for the degradation by impulsive noise

becomes difficult. Original

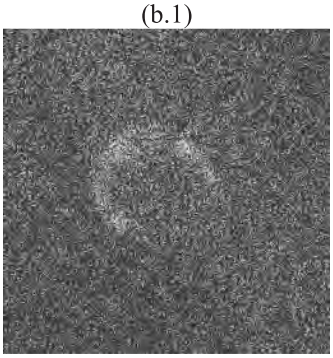
After Wavelet Shrinkage Thresholding

ER = 0.17
(a.1)

ER = 0.12, PSNR = 12.1 dB
(b.1)



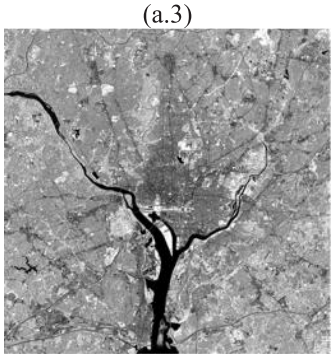
ER = 0.32
(a.2)



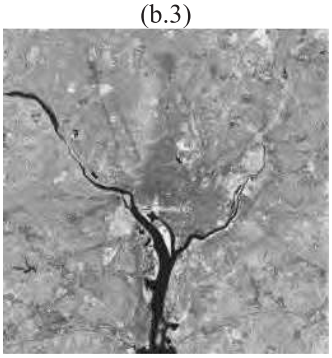
ER = 0.16, PSNR = 5.0 dB
(b.2)

ER = 0.031
(a.3)

ER = 0.021, PSNR = 17.7 dB,
(b.3)



ER = 0.021
(a.4)



ER = 0.015, PSNR = 17.0 dB
(b.4)

Figure 4.25: The examples of images that cannot be sparsified by wavelet shrinkage thresholding. (a.*i*) original image and (b.*i*) image after the wavelet shrinkage thresholding.

4.5.2 Effect of Noise Magnitude

In Figure 4.2, the higher noise magnitude led to the larger range of effective η. In this experiment, the minimum magnitude of $3y_{max}$ (which is the 2 SD distance from $\mu = 5y_{max}$) was ignored. Pepper, Mandrill, Lena, Circle and Ripple were corrupted by impulsive noise with $\mu = 3y_{max}$ and the magnitude of impulsive noise was allowed to be lower than y_{max}. Since y_{max} was 2 SD distance from μ, very few noise element (2.2% of noisy elements) would be lower than y_{max}. M/N was set to 0.40 which would lead to the best performance. Four levels of p (0.05, 0.10, 0.15 and 0.20) were investigated. Since HUBER was effective when the noise magnitude was small, the reconstruction by GBF and GSSF were compared with the one by HUBER.

Figures 4.26 - 4.30 show the reconstruction result. At $p = 0.05$, 0.10 and 0.15, the reconstruction by GBF (the second row) was the worst among the three. The reconstruction by GSSF (the third row) was better than GBF in all cases, which indicated that the slope detection was more robust. The reconstruction by HUBER (the fourth row) was much better at $p = 0.05$ and 0.10; however, the reconstruction quality was quickly deteriorated when $p = 0.15$ and 0.20.

The experiment indicated that GBF and GSSF were not effective when the noise was allowed to be smaller than y_{max}, since the drop in the ER would not be distinct. Furthermore, the disappearance of elements with large magnitude, while the large noise exists, leads to unpredictable change of ER. For small impulsive noise, HUBER was found to be the effective algorithm up until $p \leq 0.10$.

Original
(a.1)

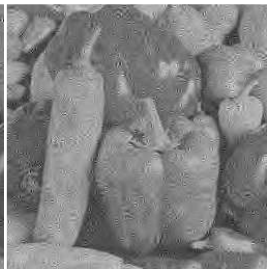
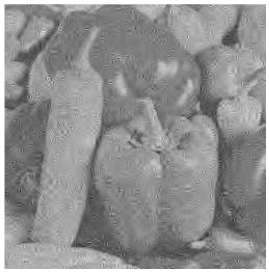
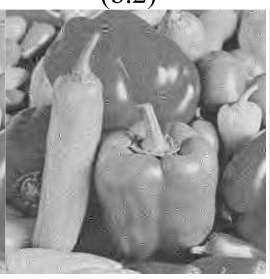
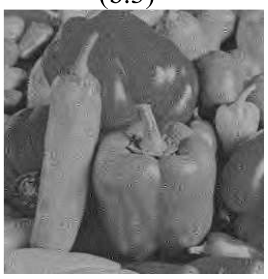
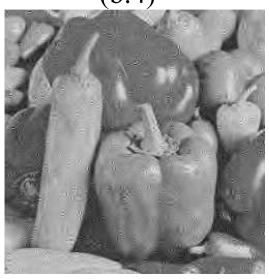
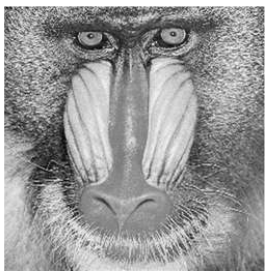
$p = 0.05$	$p = 0.10$	$p = 0.15$	$p = 0.20$
			
PSNR = 22.0 dB (b.1)	PSNR = 18.4 dB (b.2)	PSNR = 17.2 dB (b.3)	PSNR = 17.0 dB (b.4)
			
PSNR = 22.0 dB (c.1)	PSNR = 22.0 dB (c.2)	PSNR = 20.7 dB (c.3)	PSNR = 20.0 dB (c.4)
			
PSNR = 27.8 dB (d.1)	PSNR = 24.4 dB (d.2)	PSNR = 19.4 dB (d.3)	PSNR = 18.3 dB (d.4)

Figure 4.26: The reconstruction of Peppers at $M/N = 0.40$ and $\mu = 3y_{max}$. (a.1) original image, the reconstruction by (b.*i*) GBF (g= 5%), (c.*i*) GSSF (g= 10%, $c = 0.125$), (d.*i*) HUBER.



Original
(a.1)

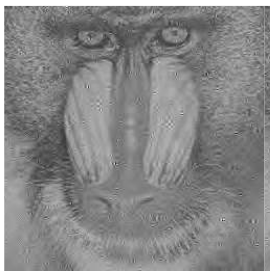
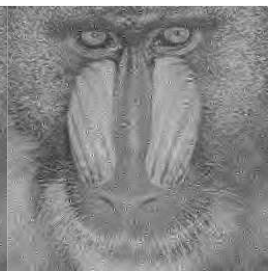
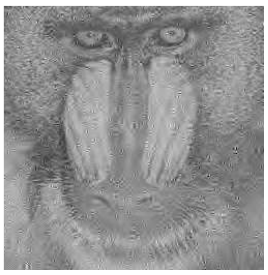
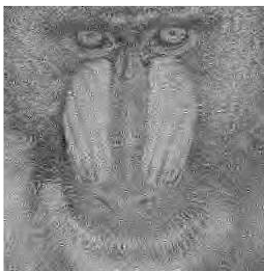
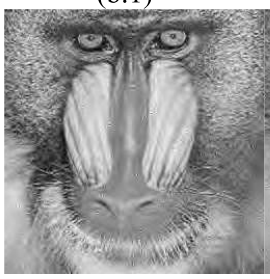
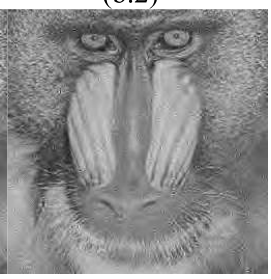
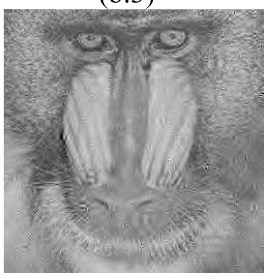
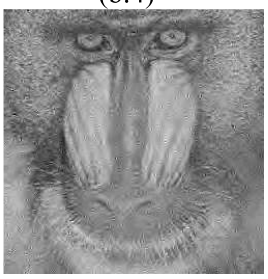
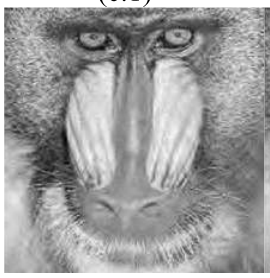
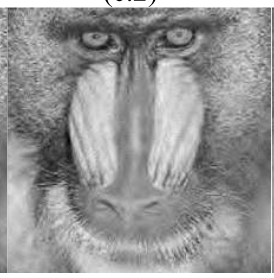
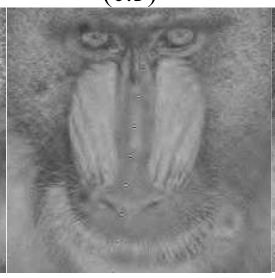
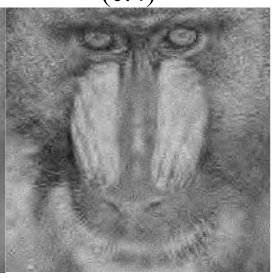
p = 0.05	p = 0.10	p = 0.15	p = 0.20
			
PSNR = 18.4 dB (b.1)	PSNR = 17.7 dB (b.2)	PSNR = 16.2 dB (b.3)	PSNR = 15.0 dB (b.4)
			
PSNR = 19.4 dB (c.1)	PSNR = 18.4 dB (c.2)	PSNR = 17.0 dB (c.3)	PSNR = 16.5 dB (c.4)
			
PSNR =19.5 dB (d.1)	PSNR = 18.4 dB (d.2)	PSNR = 15.4 dB (d.3)	PSNR = 9.1 dB (d.4)

Figure 4.27: The reconstruction of Mandrills at M/N = 0.40 and $\mu = 3y_{max}$. (a.1) original image, the reconstruction by (b.*i*) GBF (g= 5%), (c.*i*) GSSF (g= 10%, c = 0.125), (d.*i*) HUBER.

Original
(a.1)

p = 0.05	p = 0.10	p = 0.15	p = 0.20
			
PSNR = 19.6 dB (b.1)	PSNR = 18.8 dB (b.2)	PSNR = 16.5 dB (b.3)	PSNR = 15.3 dB (b.4)
			
PSNR = 25.0 dB (c.1)	PSNR = 22.6 dB (c.2)	PSNR = 21.0 dB (c.3)	PSNR = 20.5 dB (c.4)
			
PSNR =26.3 dB (d.1)	PSNR = 23.8 dB (d.2)	PSNR = 20.5 dB (d.3)	PSNR = 10.0 dB (d.4)

Figure 4.28: The reconstruction of Lenas at M/N = 0.40 and $\mu = 3y_{max}$. (a.1) original image, the reconstruction by (b.*i*) GBF (g= 5%), (c.*i*) GSSF (g= 10%, c = 0.125), (d.*i*) HUBER.

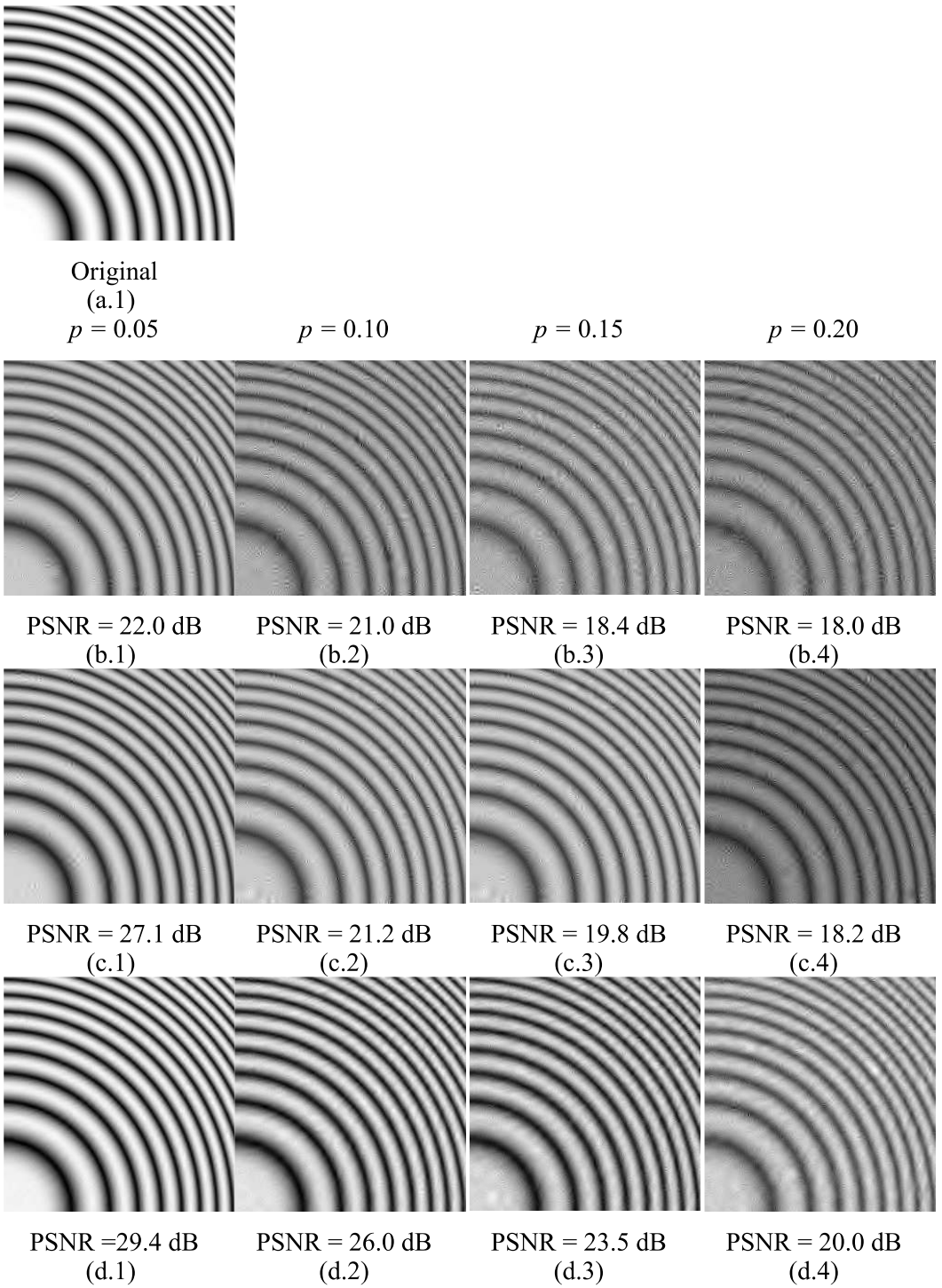
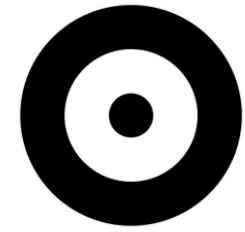


Figure 4.29: The reconstruction of Ripples at $M/N = 0.40$ and $\mu = 3y_{max}$. (a.1) original image, the reconstruction by (b.*i*) GBF (g= 5%), (c.*i*) GSSF (g= 10%, c = 0.125), (d.*i*) HUBER.



Original
(a.1)

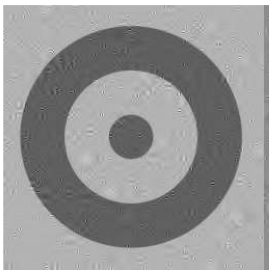
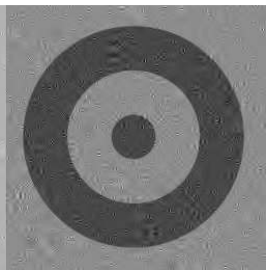
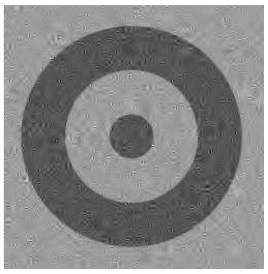
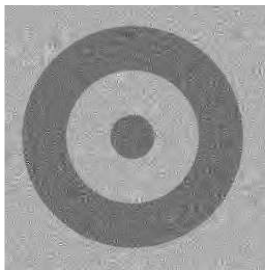
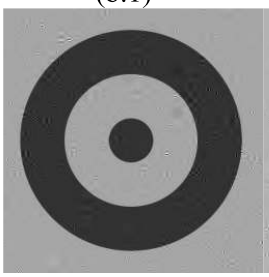
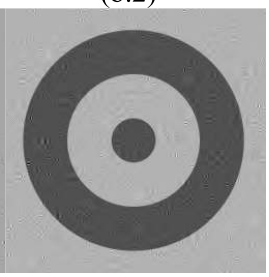
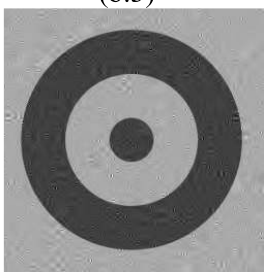
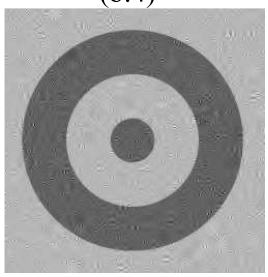
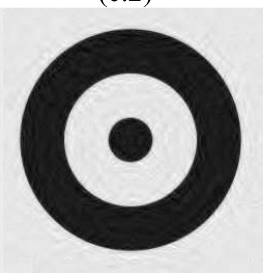
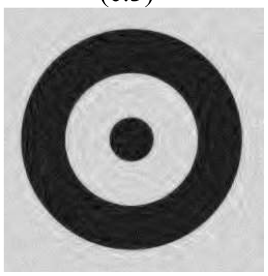
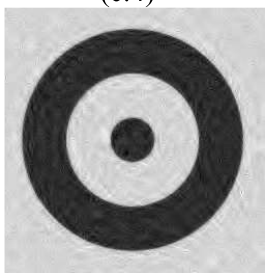
p = 0.05	p = 0.10	p = 0.15	p = 0.20
			
PSNR = 21.0 dB (b.1)	PSNR = 17.0 dB (b.2)	PSNR = 15.0 dB (b.3)	PSNR = 14.0 dB (b.4)
			
PSNR = 27.0 dB (c.1)	PSNR = 25.6 dB (c.2)	PSNR = 23.0 dB (c.3)	PSNR = 19.2 dB (c.4)
			
PSNR =30.0 dB (d.1)	PSNR = 28.0 dB (d.2)	PSNR = 25.6 dB (d.3)	PSNR = 23.0 dB (d.4)

Figure 4.30: The reconstruction of Circles at M/N = 0.40 and $\mu = 3y_{max}$. (a.1) original image, the reconstruction by (b.*i*) GBF (g= 5%), (c.*i*) GSSF (g= 10%, c = 0.125), (d.*i*) HUBER.

CHAPTER V

CONCLUSIONS

5.1 Conclusions

This thesis proposes the integration of a binary search to Approximated Measurement Preprocessing (AMP) [20] in order to reduce the computational time of AMP. Two preprocessing algorithms were proposed: the Greedy Boundary Finder (GBF) and the Greedy Steep Slope Finder (GSSF). The computational time is reduced by exploiting the redundancy characteristic of an image signal. The binary search is used to detect the number of noisy elements to within $+g$ of the actual number, where g is the predefined constant and has the unit of percent to the length of a compressed measurement signal ($\mathbf{y}$). The experiments show that GBF and GSSF provided the comparable result to AMP, while the computational time was reduced, when $\mathbf{y}$ contained a high number of noisy elements which could be caused by longer $\mathbf{y}$ or higher noise probability (p).

GBF and GSSF were compared with AMP, reconstruction by Huber penalty function (HUBER) [19] and Lorentzian Iterative Hard Thresholding (LIHT) [18]. It was found that GBF, GSSF and AMP were more optimal (provided higher PSNR at lower computational time) than HUBER and LIHT.

GBF can be considered as AMP with the binary search as the searching technique. The experiment indicated that GBF with g = 5% and the energy ratio threshold (η) = 0.1 was optimal. It should be used in place of AMP when p was higher than 0.05.

GSSF uses the change of the energy ratio (ER) to detect the presence of impulsive noise. The experiment indicated that GBF with g = 10% and the slope scale (c) < 1 is optimal. It provided the comparable PSNR to AMP but with lower computational time. Compared to GBF, GSSF provided slightly lower PSNR and

required higher computational time. The benefit of GSSF over GBF was slightly more robustness against small noise.

GBF and GSSF are not efficient in the following cases.

- The application to the image that is severely degraded by wavelet shrinkage thresholding.
- The rejection of the noise whose magnitude is lower than the maximum magnitude in the noiseless **y**.

5.2 Future Research

In Section 4.5, it was found that HUBER was a good reconstruction when the magnitude of the impulsive noise was allowed to be smaller than the maximum magnitude. However, the reconstruction with Huber regularization function is not robust to (1) high p and (2) the large noise. To increase the tolerance to the noise density and the noise magnitude, *ER* is included as *a priori* to HUBER. The regularization term from the L_1 norm of the sparse signal is replaced by the L_1 norm of the high frequency component of the sparse signal (L_1-HF). The reconstruction with L_1-HF is formulated as follows.

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \; h(\mathbf{y} - \mathbf{\Phi x}) + \alpha \left\| \mathbf{x}_{hf} \right\|_1, \tag{5.1}$$

where $\mathbf{x}_{hf}$ is the vector containing elements outside LL_3 subband;
$h(\cdot)$ is the Huber penalty function and defined as (2.13).
α is the regularization parameter;
$\|\cdot\|_1$ is the L_1 norm.

The preliminary comparison between the two regularization terms was performed on Pepper (Figure 5.1) and Mandrill (Figure 5.2). The figures clearly show that the change in the regularization term improved the tolerance of the noise magnitude (in term of the magnitude mean (μ)) and p. However, the computational time of HUBER is high and should be reduced. Furthermore, though *a priori* increases the robustness of HUBER, its weakness to large noise is not completely

solved, because the weakness is caused by the unbound characteristic of Huber function. Thus, the integration of L_1-HF with GBF (as well as GSSF) should be investigated in order to create the rejection that is robust to all noise levels.

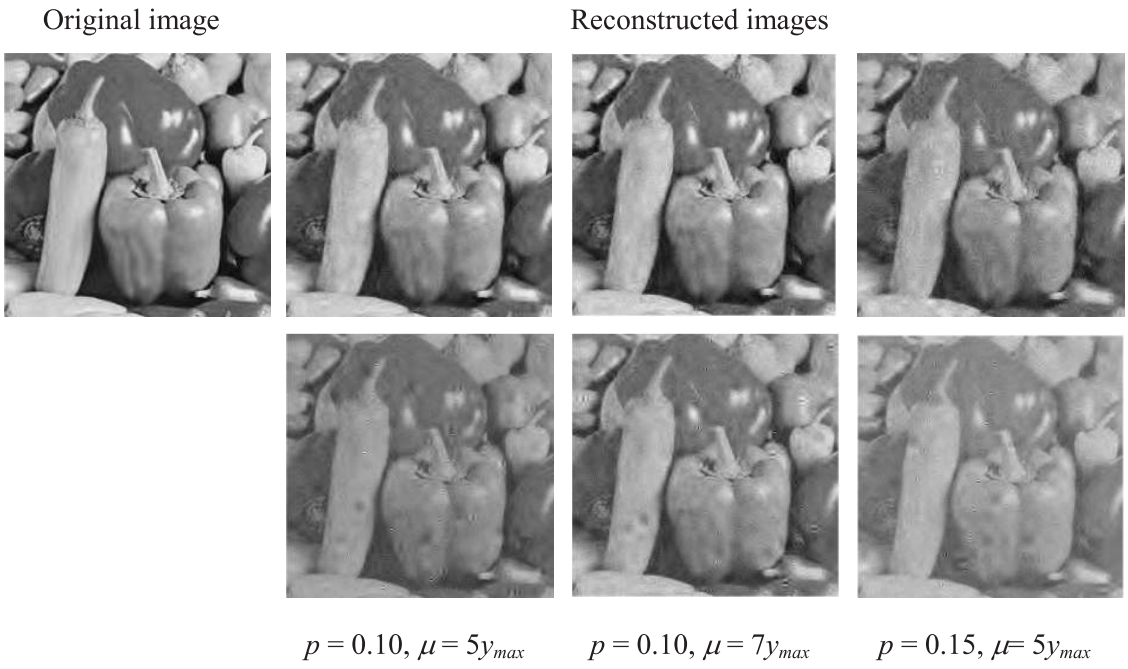


Figure 5.1: The reconstruction of Peppers at different p's and magnitude means (μ). The top and the bottom rows of the reconstructed images show the results of of HUBER with L_1-HF norm and L_1 norm, respectively.

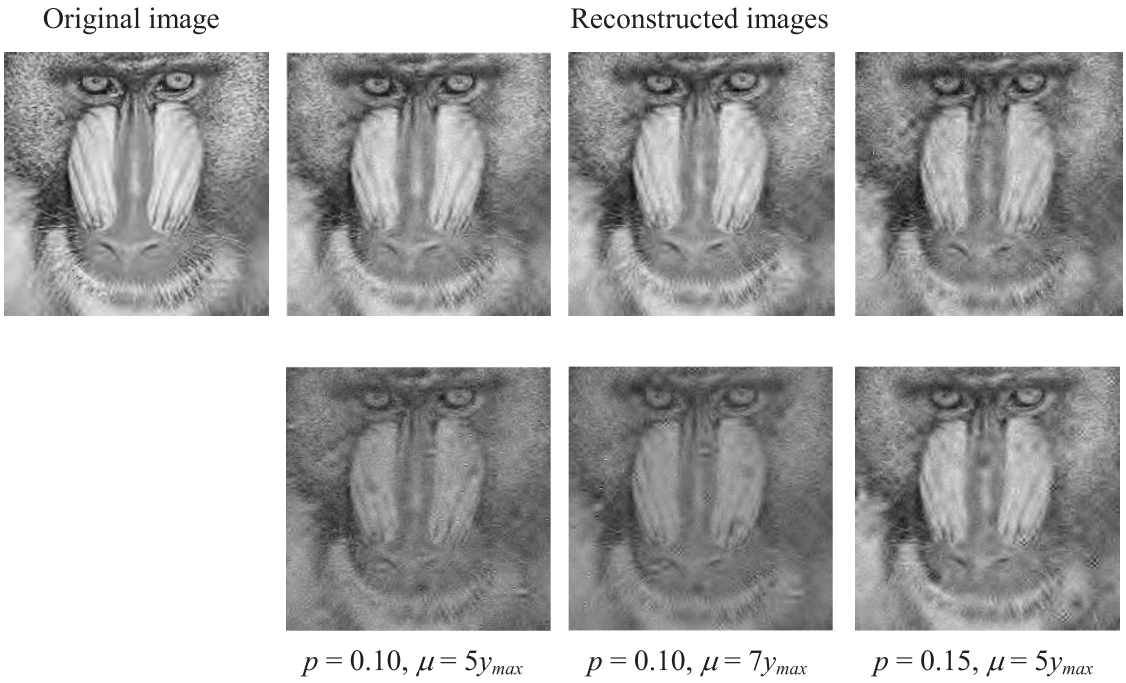


Figure 5.2: The reconstruction of Mandrill at different p's and magnitude means. The top and the bottom rows of the reconstructed images show the results of of HUBER with L_1-HF norm and L_1 norm, respectively.

REFERENCES

[1] Donoho, D. L. Compressive sensing. IEEE Trans. Inf. Theory 52 (4) (2006) : 1289-1306.

[2] Candes, E. J., and Romberg, J. Sparsity and incoherence in compressive sampling. Inverse Problem 223 (3) (2007) : 969-985.

[3] Candes, E. J., and Wakin, M. B. An introduction to compressive sampling. IEEE Signal Processing Mag. 25 (March 2008) : 21-30.

[4] Arguello, H., and Arce, G. R. Code aperture optimization for spectrally agile compressive imaging. Journal of the Optical Society of America, 28 (11) (2011) : 2400-2413.

[5] Shi, G., Gao, D., Liu, D., and Wang, L. High resolution image reconstruction: a new imager via movable random exposure. In Proc. ICIP, (2009) :1177-1180.

[6] Lustig, M., Donoho, D., and Pauly, J. M. Sparse MRI: The application of compressed sensing for rapid MR imaging. Journal of Magnetic Resonance in Medicine, 58 (6) (December 2007) : 1182-1195.

[7] Lustig, M., Santos, J. M., Donoho, D., and Pauly, J. M. k-t SPARSE: High frame rate dynamic MRI exploiting spatio-temporal sparsity. in ISMRM, Seattle, Washington, (May 2006)

[8] Chen, S., Donoho, D. L., and Saunders, M. Atomic decomposition by basis pursuit. SIAM Review 43 (1) (2001) : 129-159.

[9] Tibshirani, R. Regression shrinkage and selection via the lasso. Journal of Royal Statistical Society 58 (1) (1996) : 267–288.

[10] Mallat, S. G., and Zhang, Z. Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. 41 (12) (1993) : 3397-3415.

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

[12] Carrillo, R. E., Polania, L. F., and Barner, K. E. Iterative algorithm for compressed sensing with partially know support. In Proc. IEEE ICASSP (2010) : 3654-3657.

[13] Baraniuk, R.G., Cevher, V., Duarte, M.F., and Hegde, C. Model-based compressed sensing. IEEE Trans. Inf. Theory 56 (4) (2010): 1982-2001.

[14] Blumensath, T., and Davies, M. E. Normalized iterative hard thresholding: guaranteed stability and performance. IEEE. J. Sel. Topics Sigal Process 4 (2) (2010) : 298-309.

[15] Candès, E. J., and Tao, T. The dantzig selector: statistical estimation when *p* is much larger than *n*. Ann. Statist. 35 (6) (2007) : 2313-2351.

[16] Carrillo, R. E., Aysal, T. C., and Barner, K. E. A Generalized Cauchy distribution framework for problems requiring robust behavior. EURASIP J. on Advance in Sigal Processing 2010 (2010) : Article ID 312989. 19 pages.

[17] Carrillo, R. E., Barner, K. E., and Aysal, T. C. Robust sampling and reconstruction methods for sparse signals in the presence of impulsive noise. IEEE J. Sel. Topics Signal Process 4 (2) (2010) : 392-408.

[18] Carrillo, R. E., and Barner, K. E. Lorentzian based iterative hard thresholding for compressed sensing. In Proc. IEEE ICASSP (2011) : 3664-3667.

[19] Pham, D., and Venkatesh, S. Improved image recovery from compressed data contaminated with impulsive noise IEEE Trans. Inf. Theory 21 (1) (2012) : 397-405.

[20] P. Sermwuthisarn, D. Gansawat, V. Patanavijit and S. Auethavekiat. Impulsive noise rejection method for compressed measurement signal in compressed

sensing. EURASIP J. on Advances in Signal Processing 2012 (2012) doi:10.1186/16876180201268.

[21] Schmidt, A., and Moura, J. M. F. Distributed field reconstruction with model-robust basis pursuit. in Proc. IEEE ICASSP, (2012) : 2673-2676.

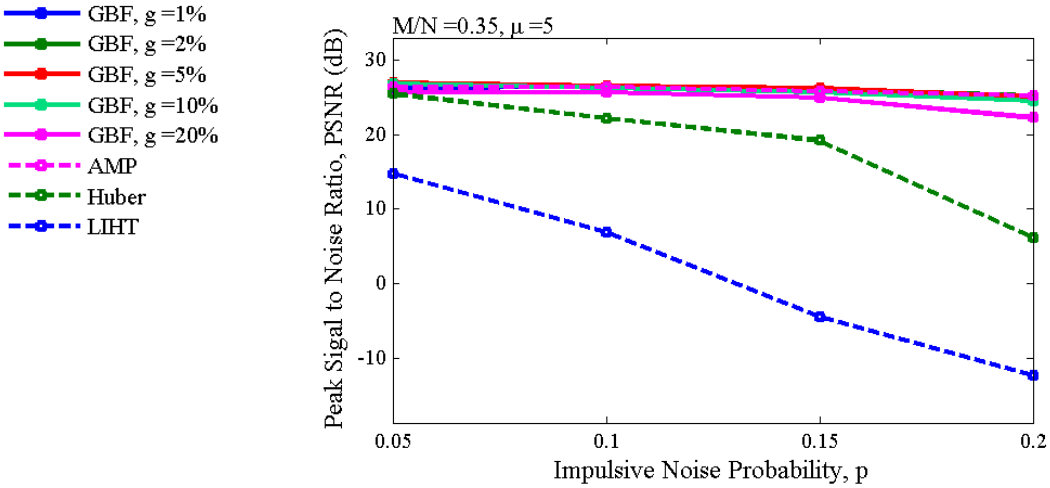
[22] Donoho, D. L. De-noising by soft thresholding. IEEE Trans. Inf. Theory 38 (2) (1995) : 613-627.

APPENDICES

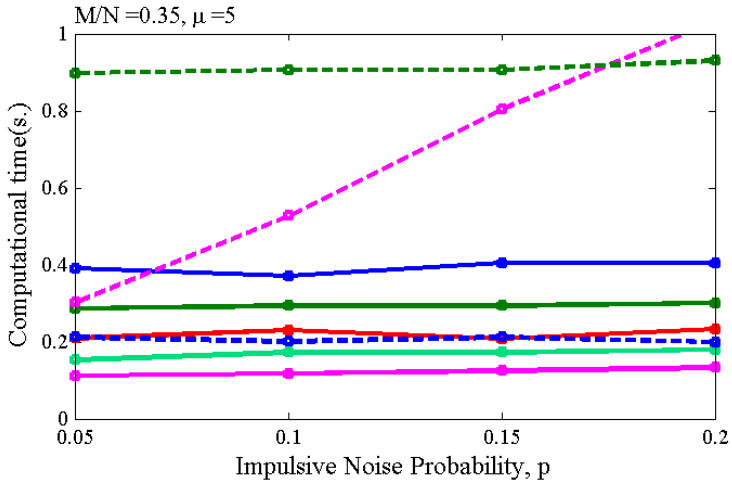
APPENDIX A

Plots of Performance Comparison

A.1 Greedy Boundary Finder (GBF) at M/N = 0.35 and 0.40

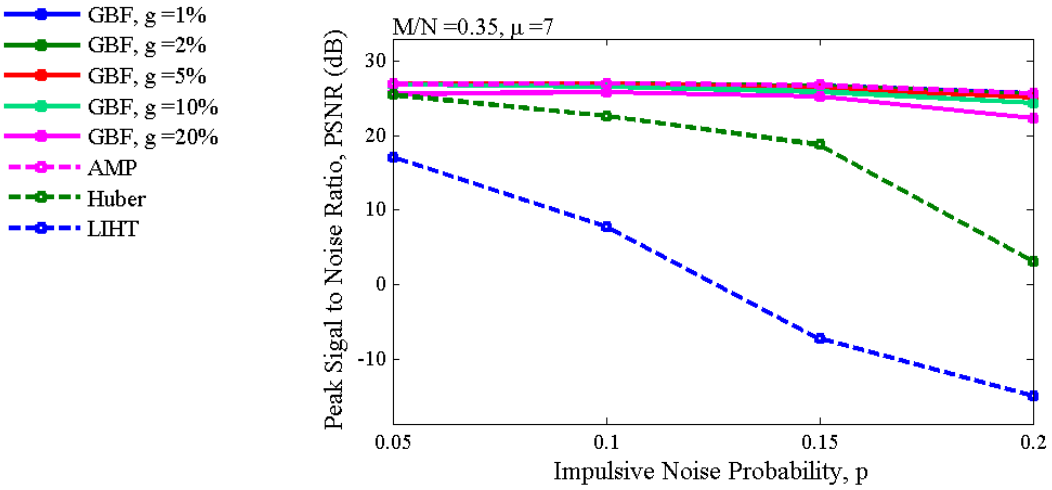


(a)

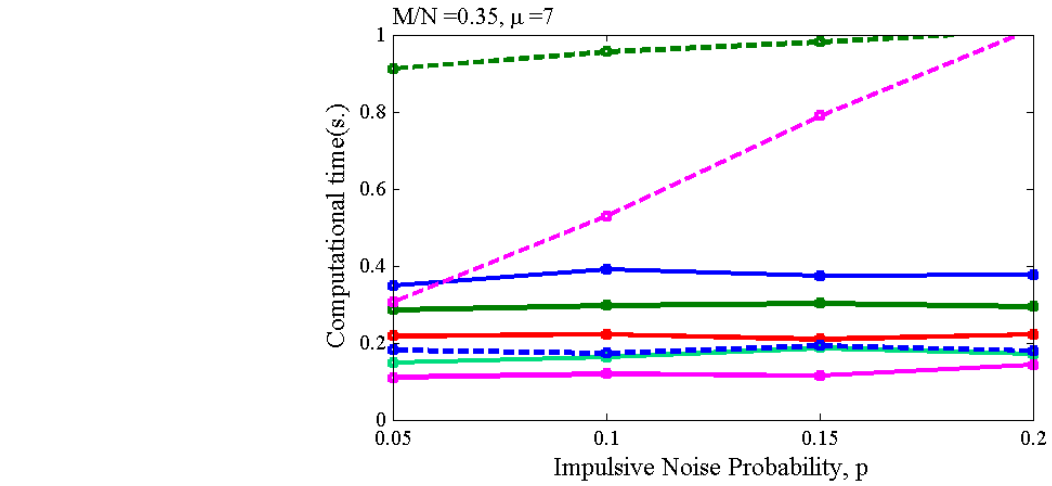


(b)

Figure A.1: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.35, and μ=5y_{max}. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.



(a)



(b)

Figure A.2: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.35, and $\mu=7y_{max}$. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

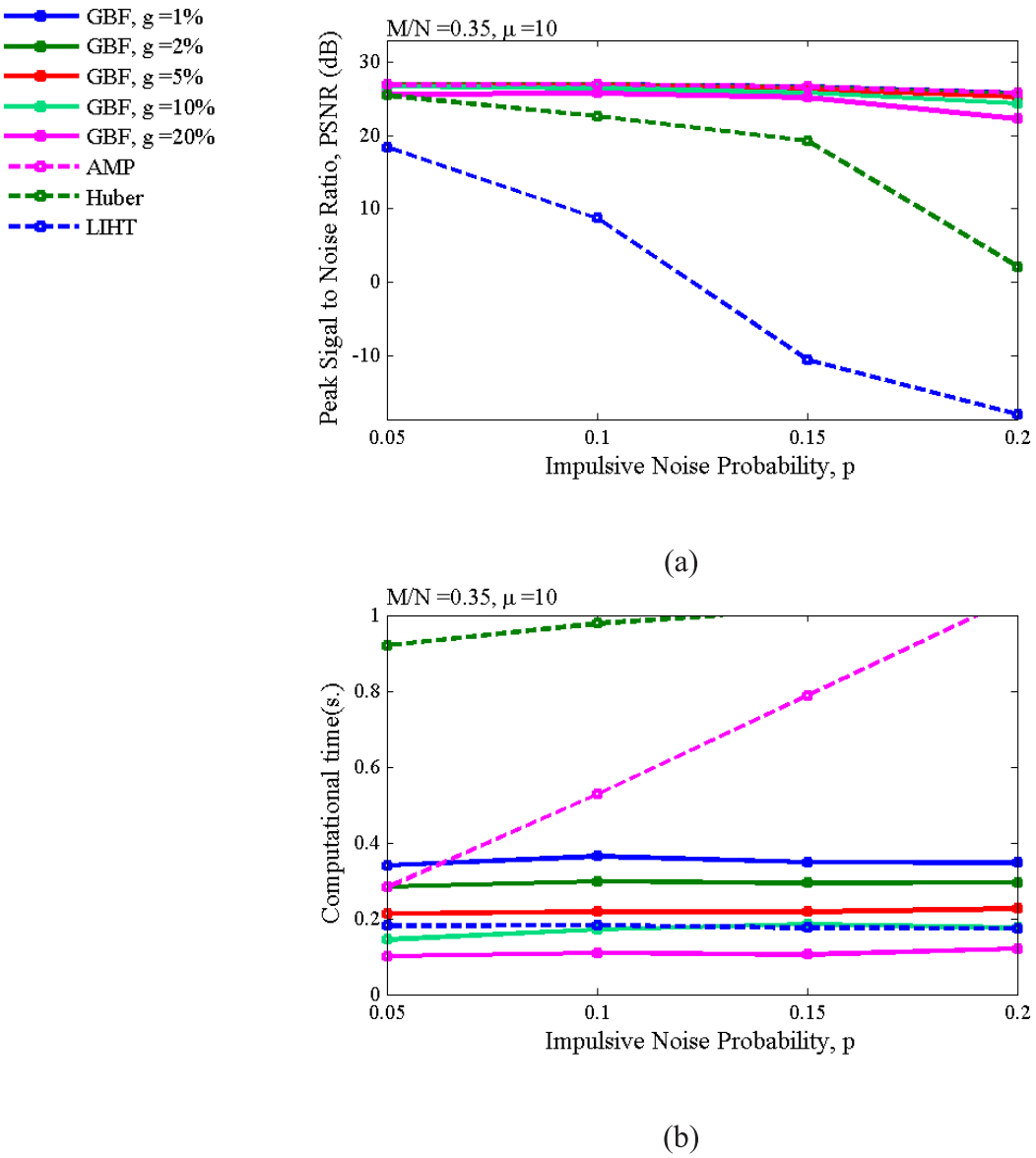


Figure A.3: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.35, and $\mu=10y_{max}$. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

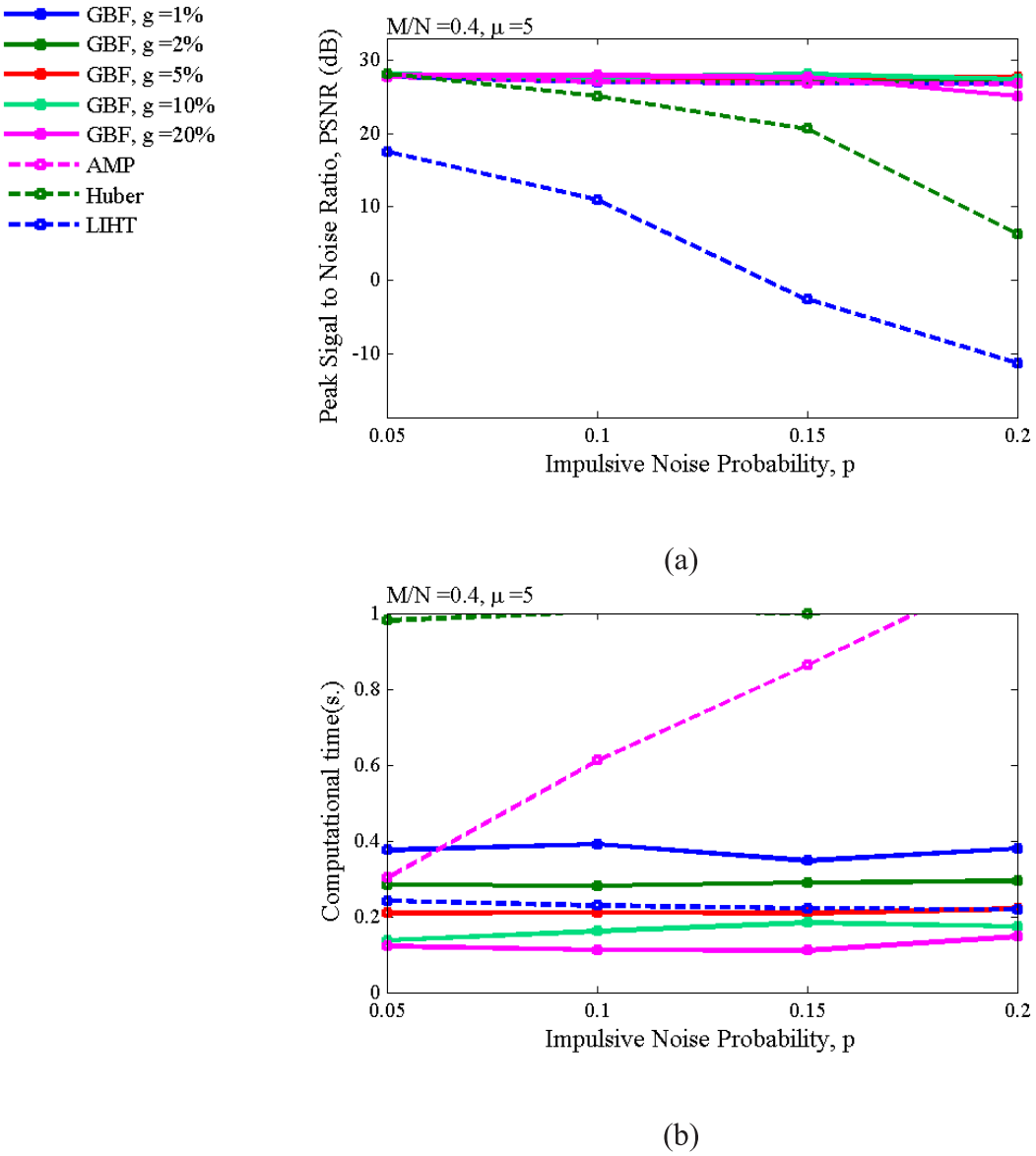
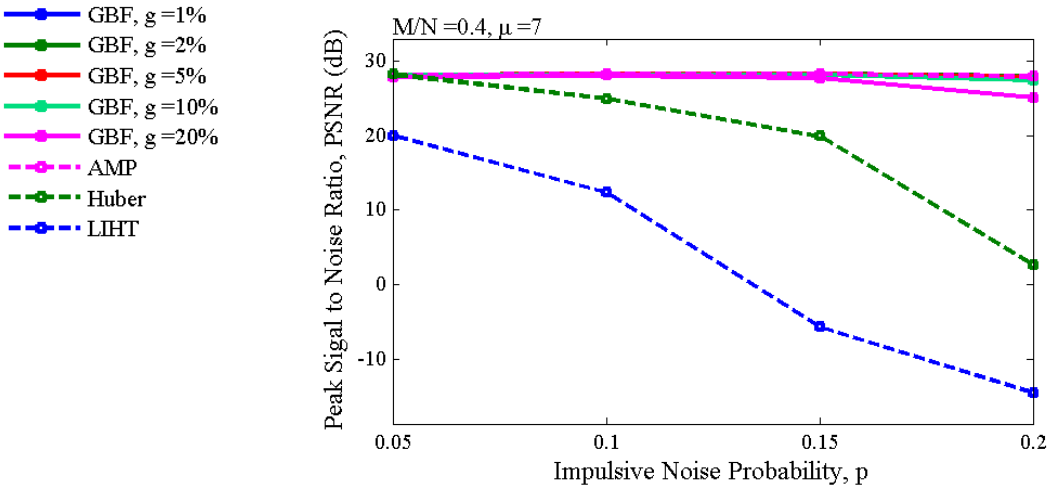
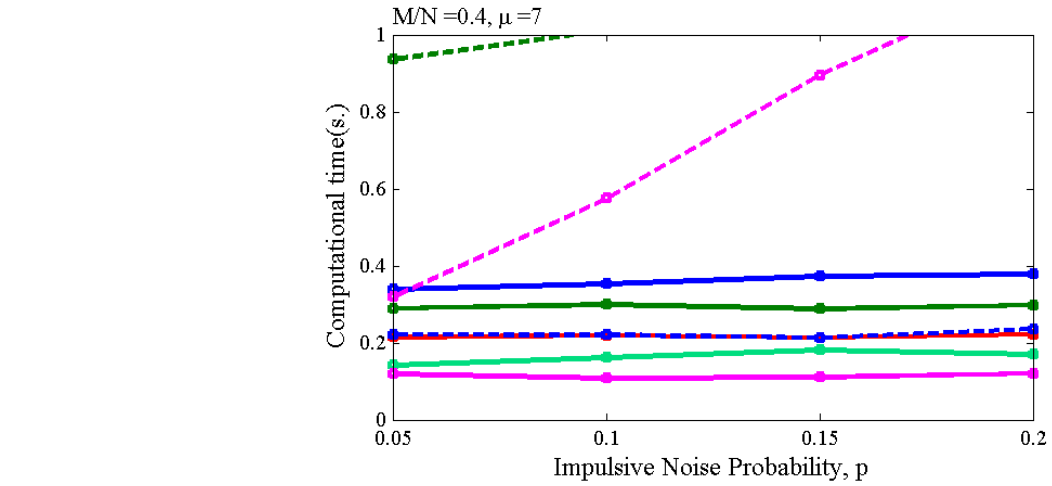


Figure A.4: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.40, and $\mu=5y_{max}$. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

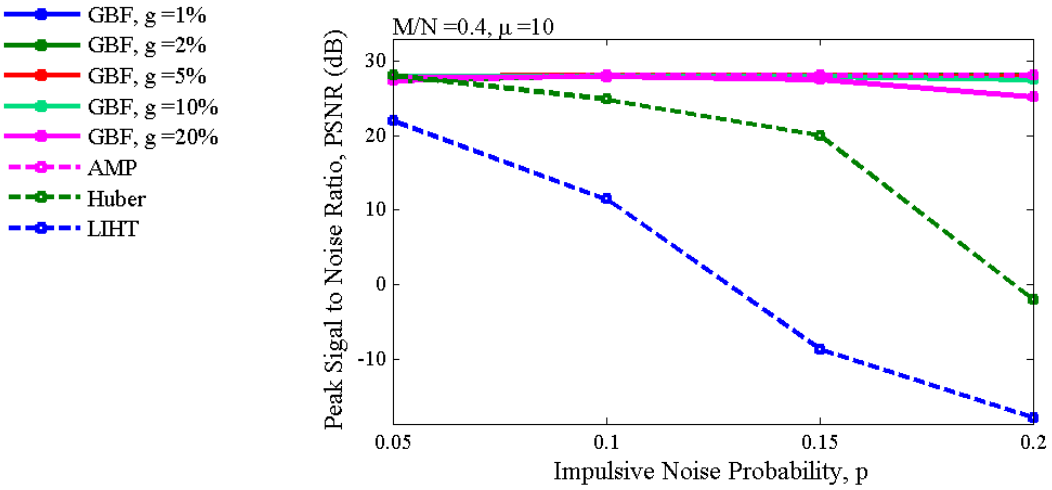


(a)

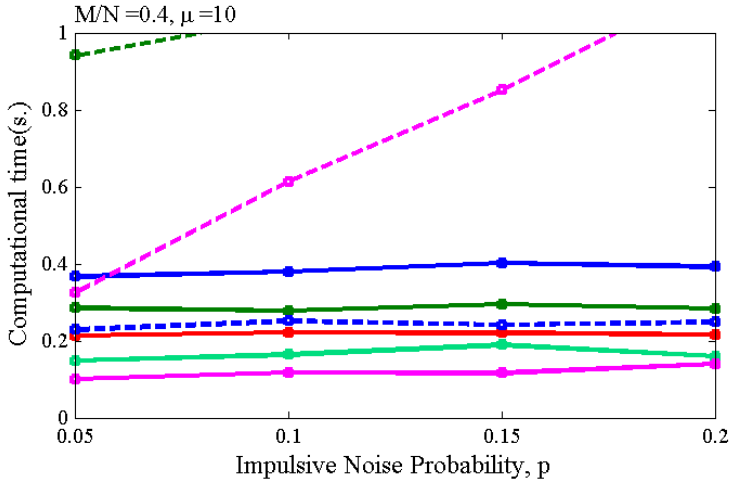


(b)

Figure A.5: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.40, and $\mu=7y_{max}$. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.



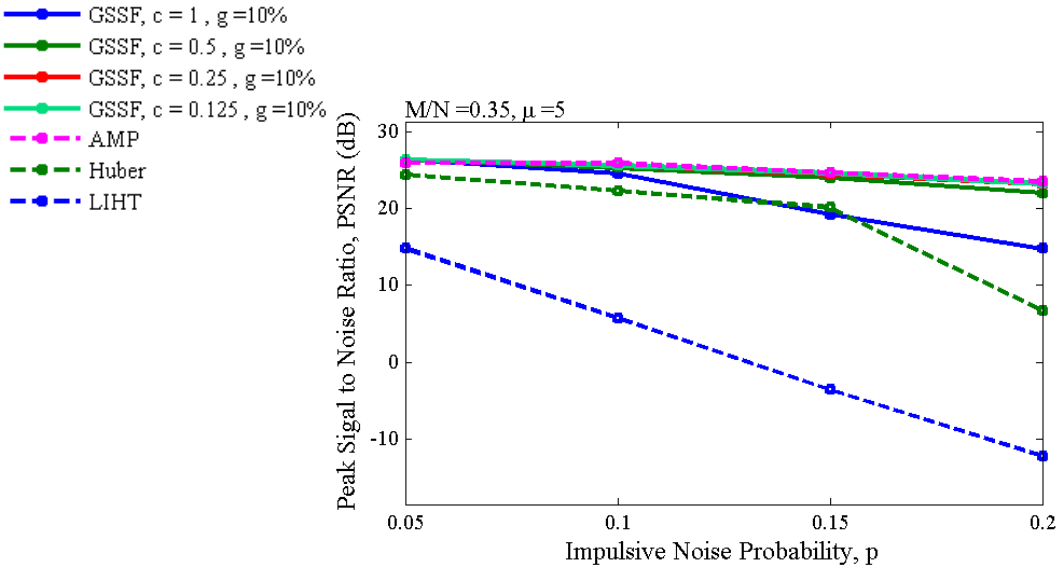
(a)



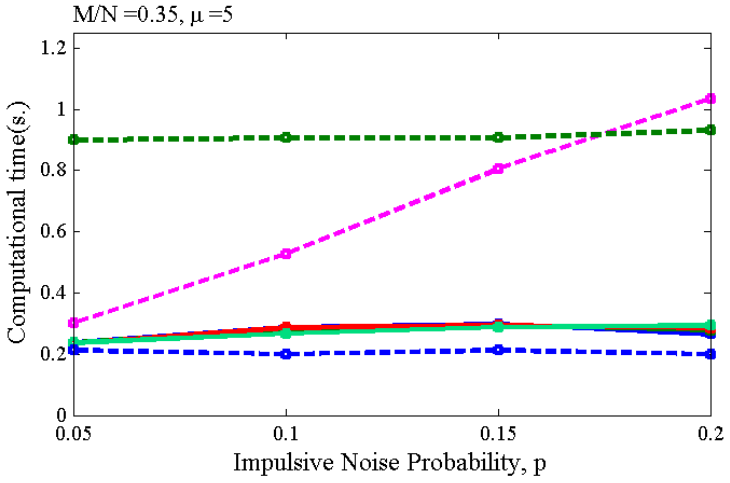
(b)

Figure A.6: Performance comparison between GBF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) when M/N =0.40, and μ=10y_{max}. (a) average PSNR and (b) average computational time (per block). g is the gap resolution in GBF.

A.2 Greedy Steep Slope Finder (GSSF) at M/N = 0.35 and 0.40

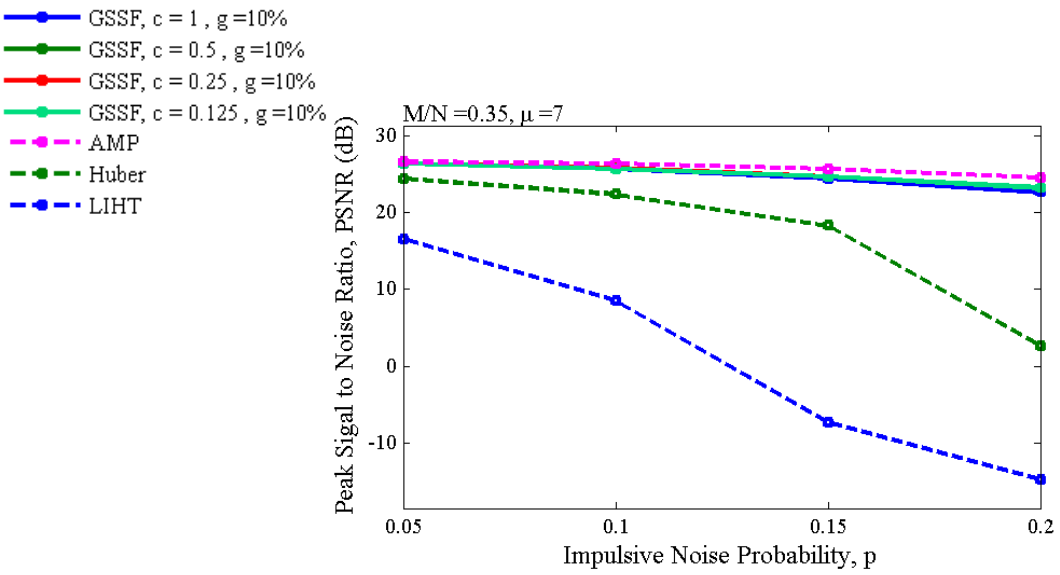


(a)

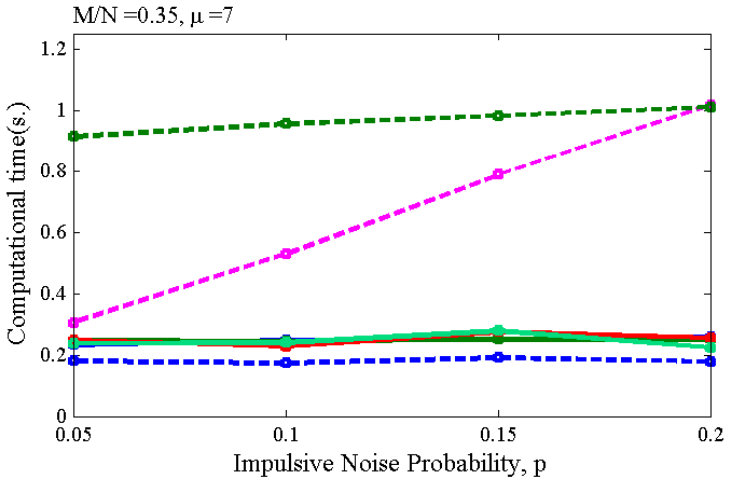


(b)

Figure A.7: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.35, and $\mu = 5y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

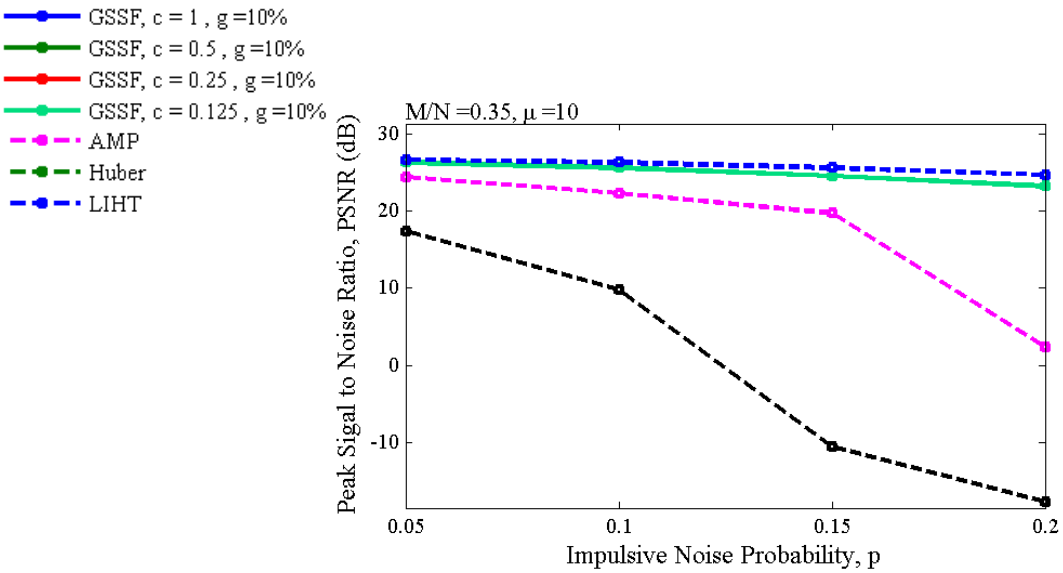


(a)

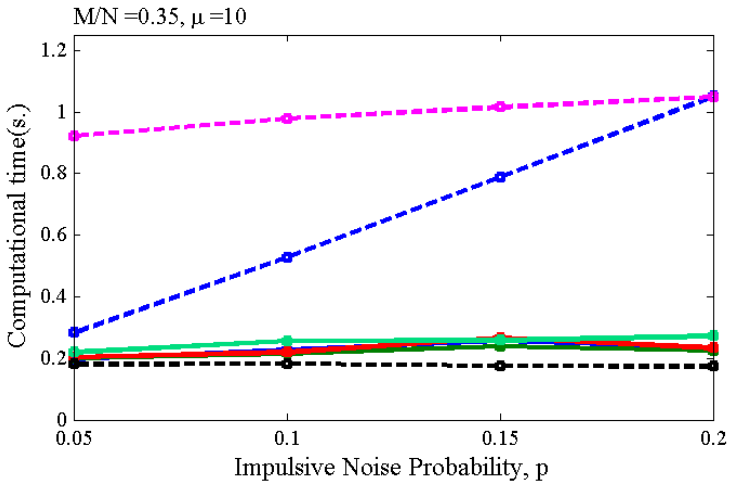


(b)

Figure A.8: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.35, and $\mu = 7y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

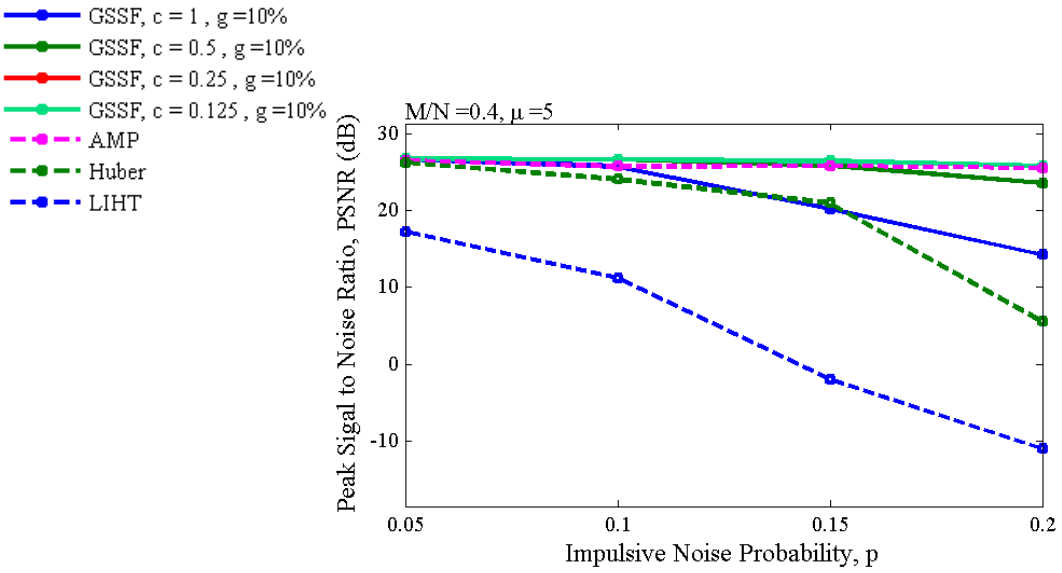


(a)

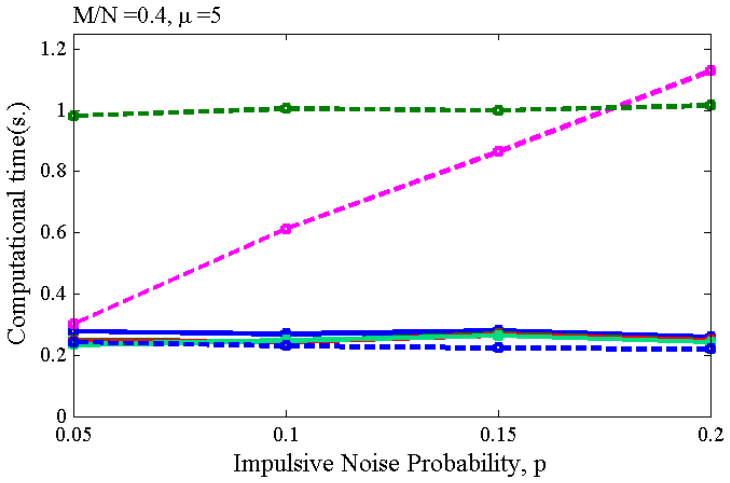


(b)

Figure A.9: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.35, and $\mu = 10y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

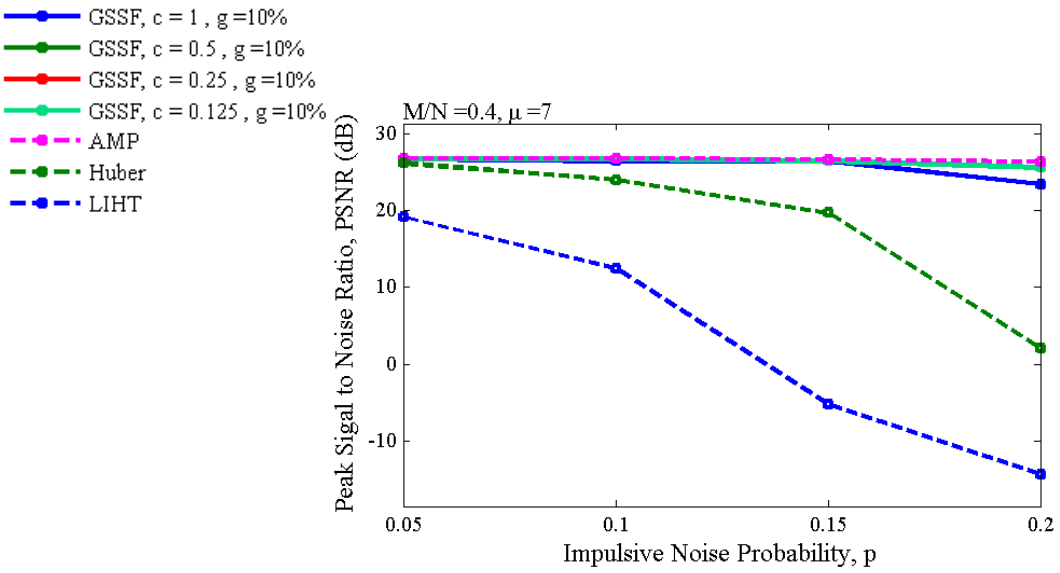


(a)

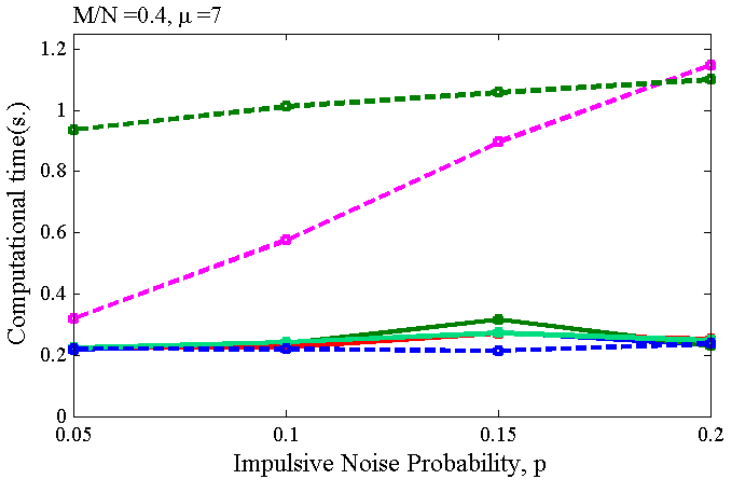


(b)

Figure A.10: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.40, and $\mu = 5y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

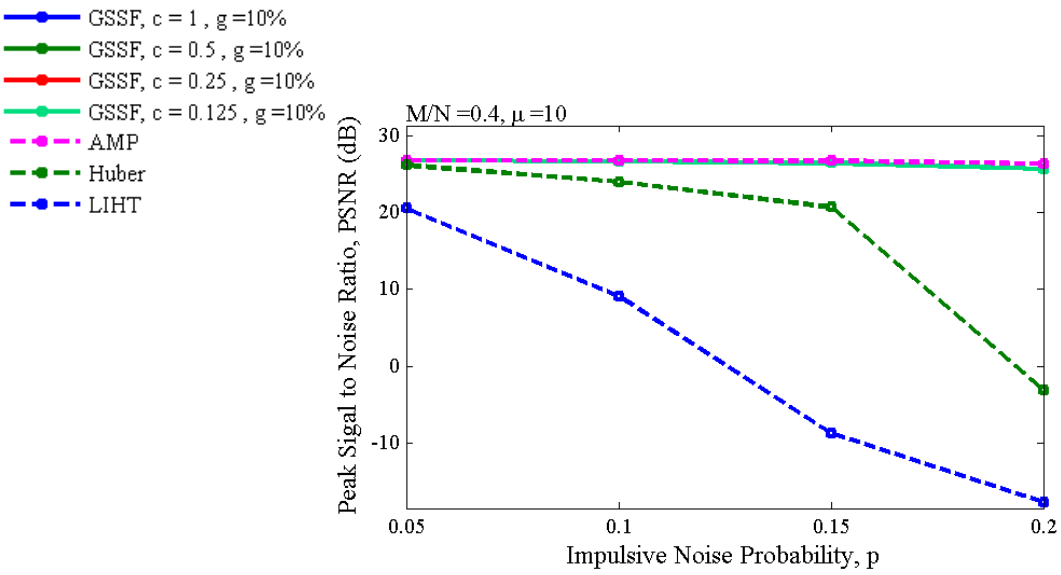


(a)

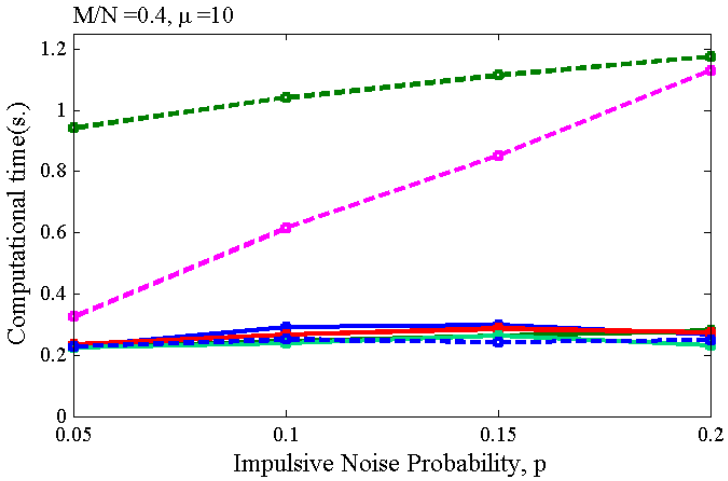


(b)

Figure A.11: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.40, and $\mu = 7y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.



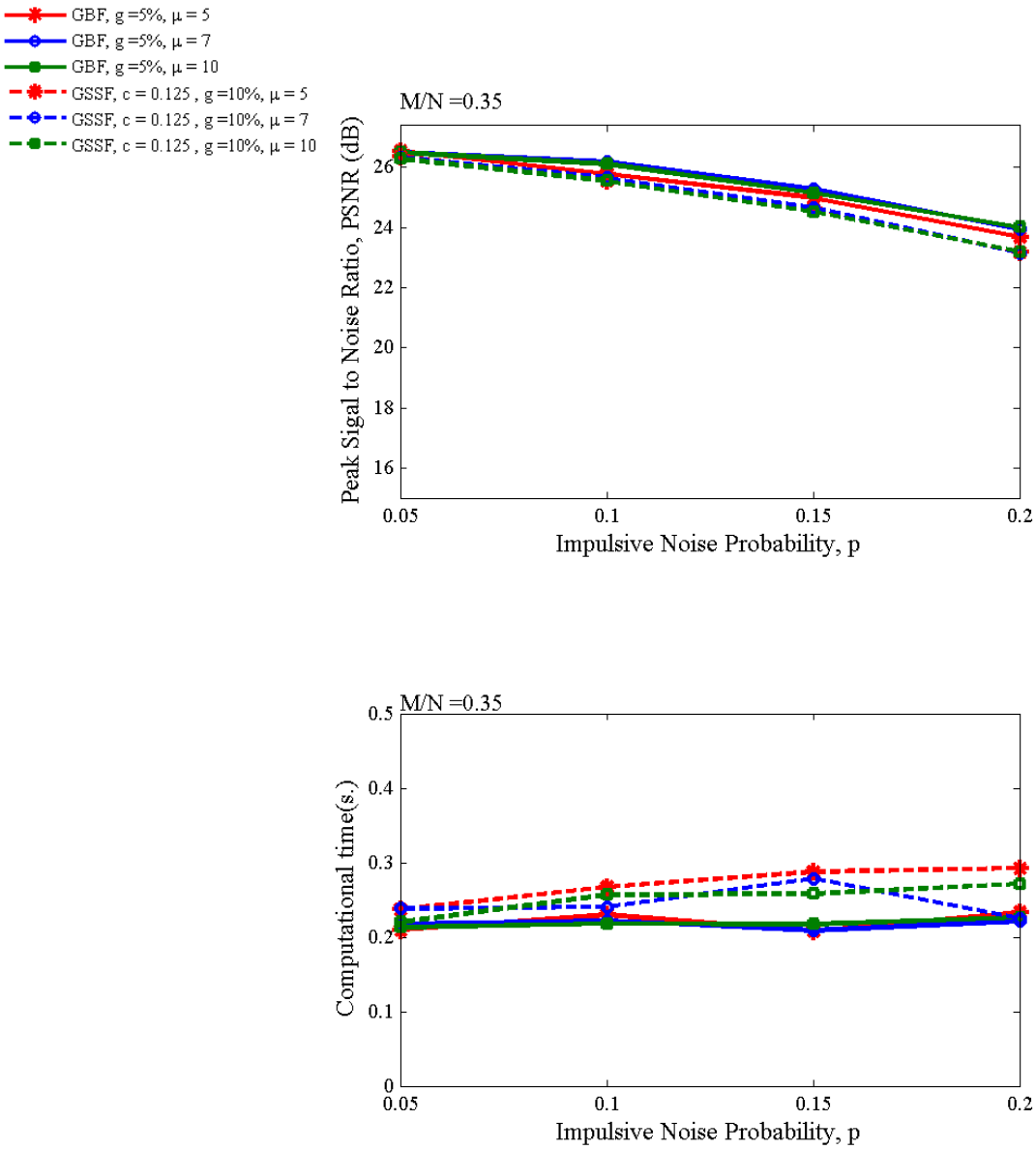
(a)



(b)

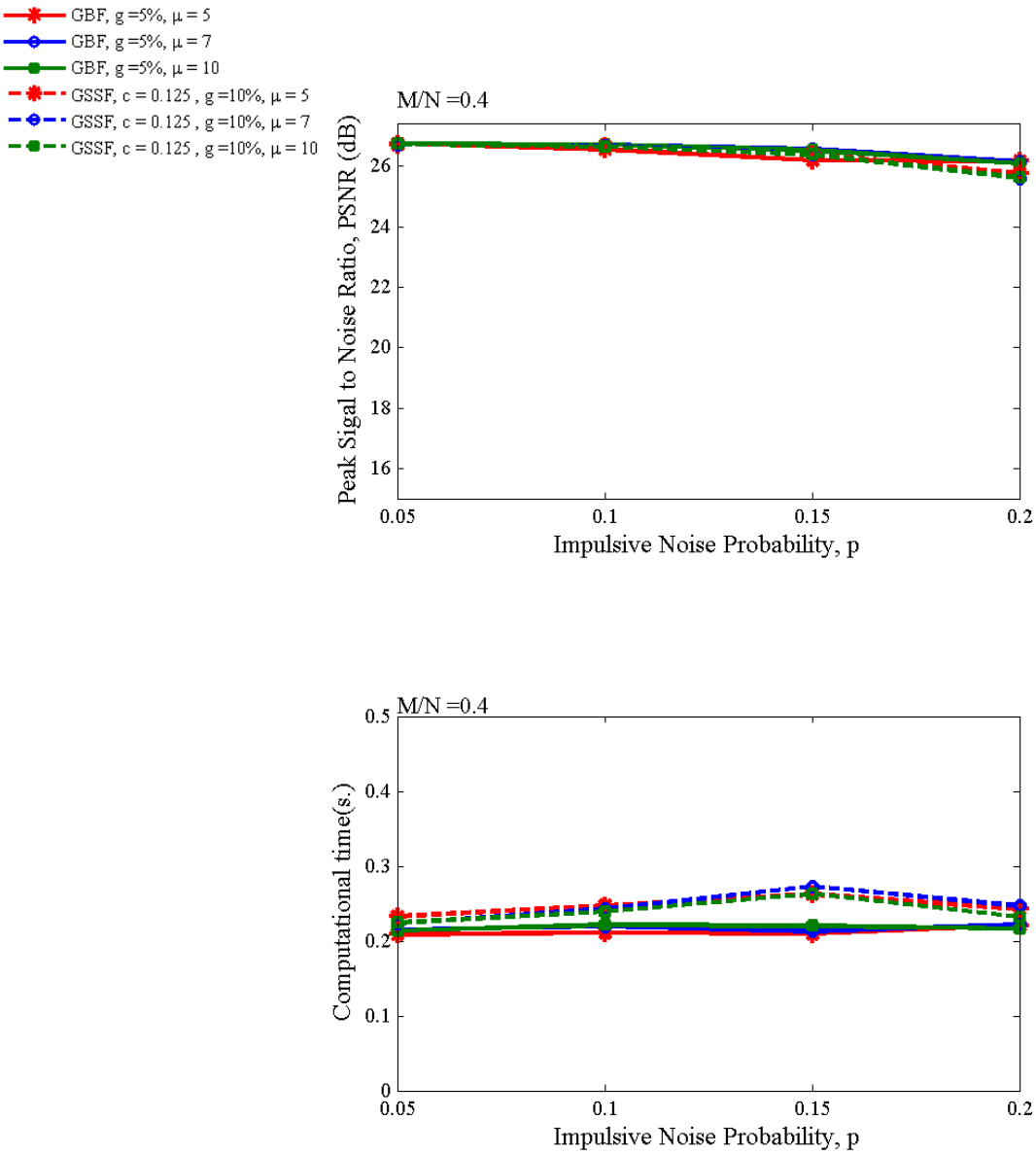
Figure A.12: Performance comparison between GSSF (bold line), AMP (dashed magenta line), HUBER (dashed green line) and LIHT (dashed blue line) at M/N = 0.40, and $\mu = 10y_{max}$. (a) average PSNR and (b) average computational time (per block). g and c are the gap resolution and slope scale in GSSF.

A.3 Comparative Evaluation Between GBF and GSSF at *M*/*N* = 0.35 and 0.40



(b)

Figure A.13: Performance comparison between GBF (g = 5%, bold line), and GSSFs (g = 10%, c = 0.125, dashed line) when M/N = 0.35. (a) average PSNR and (b) average computational time (per block).



(b)

Figure A.14: Performance comparison between GBF (g = 5%, bold line), and GSSFs (g = 10%, c = 0.125, dashed line) when M/N = 0.40. (a) average PSNR and (b) average computational time (per block).

APPENDIX B

Publications and Presentations

[1] Suwichaya Suwanwimolkul, Parichat Sermwuthisarn, Supatana Auethavekiat,

"Greedy Gap's Boundary Finder: the Impulsive Noise Rejection for Compressed Measurement Image Signal," Accepted for the 12th IEEE International Symposium on Communications and Information Technologies (ISCIT 2012), Gold Coast, Australia, Oct. 2 - 5, 2012.

[2] Suwichaya Suwanwimolkul, Parichat Sermwuthisarn, Supatana Auethavekiat,

"Greedy Steep Slope Finder: the Fast Impulsive Noise Rejection for Compressed Measurement Image Signals," Accepted for IEEE International Symposium on Intelligent Signal Processing and Communication Systems 2012 (ISPACS 2012), New Taipei, Taiwan, Nov. 4 - 7, 2012.

[3] Suwichaya Suwanwimolkul, Jitkomut Songsiri, Parichat Sermwuthisarn, Supatana Auethavekiat,

"L_1 Norm of High Frequency Components as a Regularization Term for Compressed Sensing Reconstruction of Image Signals," Accepted for IEEE International Symposium on Intelligent Signal Processing and Communication Systems 2012 (ISPACS 2012), New Taipei, Taiwan, Nov. 4 - 7, 2012.

VITAE

Suwichaya Suwanwimolkul was born in Bangkok, Thailand, in 1988. She received her Bachelor’s in Electrical Engineering from King Mongkut's University Technology of North Bangkok, Thailand, in 2011. She has been pursuing the Master degree in Electrical Engineering at Chulalongkorn University, Bangkok, Thailand, since 2011. Her research areas are compressed sensing and digital image processing.