Topics in image recovery and image quality assessment /Cui Lei.

Lei Cui

Follow this and additional works at: https://repository.hkbu.edu.hk/etd_oa

Recommended Citation
Cui, Lei, "Topics in image recovery and image quality assessment /Cui Lei." (2016). Open Access Theses and Dissertations. 368.
https://repository.hkbu.edu.hk/etd_oa/368

This Thesis is brought to you for free and open access by the Electronic Theses and Dissertations at HKBU Institutional Repository. It has been accepted for inclusion in Open Access Theses and Dissertations by an authorized administrator of HKBU Institutional Repository. For more information, please contact repository@hkbu.edu.hk.
HONG KONG BAPTIST UNIVERSITY
Doctor of Philosophy
THESIS ACCEPTANCE

DATE: November 16, 2016

STUDENT'S NAME: CUI Lei

THESIS TITLE: Topics in Image Recovery and Image Quality Assessment

This is to certify that the above student's thesis has been examined by the following panel members and has received full approval for acceptance in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Chairman: Dr. Cheung Kwok Wai
Associate Professor, Department of Computer Science, HKBU
(Designated by Dean of Faculty of Science)

Internal Members: Dr. Tong Tiejun
Associate Professor, Department of Mathematics, HKBU
(Designated by Head of Department of Mathematics)

Prof. Liao Lizhi
Professor, Department of Mathematics, HKBU

External Members: Prof. Chan Hon Fu Raymond
Chairman and Professor
Department of Mathematics
The Chinese University of Hong Kong

Dr. Dong Yiqiu
Associate Professor
Department of Applied Mathematics and Computer Science
Technical University of Denmark
Denmark

In-attendance: Dr. Zeng Tieyong
Associate Professor, Department of Mathematics, HKBU

Issued by Graduate School, HKBU
Topics in Image Recovery and Image Quality Assessment

CUI Lei

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

Principal Supervisor: Dr. Zeng Tieyong

Hong Kong Baptist University

November 2016
DECLARATION

I hereby declare that this thesis represents my own work which has been done after registration for the degree of PhD at Hong Kong Baptist University, and has not been previously included in a thesis or dissertation submitted to this or other institution for a degree, diploma or other qualifications.

I have read the University’s current research ethics guidelines, and accept responsibility for the conduct of the procedures in accordance with the University’s Committee on the Use of Human/Animal Subjects in Teaching and Research (HASC). I have attempted to identify all the risks related to this research that may arise in conducting this research, obtained the relevant ethical and/or safety approval (where applicable), and acknowledged my obligations and the rights of the participants.

Signature: Cui Lei
Date: November 2016
Abstract

Image recovery, especially image denoising and deblurring is widely studied during the last decades. Variational models can well preserve edges of images while restoring images from noise and blur. Some variational models are non-convex. For the moment, the methods for non-convex optimization are limited. This thesis finds new non-convex optimizing method called difference of convex algorithm (DCA) for solving different variational models for various kinds of noise removal problems. For imaging system, noise appeared in images can show different kinds of distribution due to the different imaging environment and imaging technique. Here we show how to apply DCA to Rician noise removal and Cauchy noise removal. The performance of our experiments demonstrates that our proposed non-convex algorithms outperform the existed ones by better PSNR and less computation time. The progress made by our new method can improve the precision of diagnostic technique by reducing Rician noise more efficiently and can improve the synthetic aperture radar imaging precision by reducing Cauchy noise within.

When applying variational models to image denoising and deblurring, a significant subject is to choose the regularization parameters. Few methods have been proposed for regularization parameter selection for the moment. The numerical algorithms of existed methods for parameter selection are either complicated or implicit. In order to find a more efficient and easier way to estimate regularization parameters, we create a new image quality sharpness metric called SQ-Index which is based on the theory of Global Phase Coherence. The new metric can be used for estimating parameters for a various of variational models, but also can estimate the noise intensity based on special models. In our experiments, we show the noise estimation performance with this new metric. Moreover, extensive experiments are made for dealing with image denoising and deblurring under different kinds of noise and blur. The numerical results show the robust performance of image restoration by applying our metric to parameter selection for different variational models.
Keywords: DCA, total variation, primal-dual, SQ-Index, regularization.
Acknowledgements

First, I would like to express my deepest gratitude to my supervisor Dr. Zeng Tieyong, for his constant encouragement and guidance. His patient instruction for my study and advice for my life will always be useful for my future. He always helps me and encourages me for my academic courses and my research. He is also a model for me. Every time, I ask questions, he always answers me in a short time. His serious attitude towards scientific research always encourages me to work as hard as I can.

I also want to show my gratitude to Prof.Ng, Michael Kwok-Po, C Wing Yeung and Claudia Chui who give me a lot of support and advice for my studies. Moreover, I would like to thank Mrs Chen Liyuan, Ms Zhu Hong and Mr Li Zhi, for their help. Without the discussion with them, I can not learn as much as now.

Furthermore, I would like to thank all the faculties for their useful courses and thank Mr Lo Kam Fai for his administrative instruction.

Finally, I wish to express my gratitude to my families: my parents, my husband, the parents of my husband, my lovely kids, for their love and confidence in me all through these three years.
Table of Contents

Declaration i

Abstract ii

Acknowledgements iv

Table of Contents v

List of Tables ix

List of Figures x

Chapter 1 Introduction 1

1.1 Introduction of image denoising . . . . . . . . . . . . . . . . . . . . 2
1.2 Introduction of image deblurring . . . . . . . . . . . . . . . . . . . . 3
1.3 Introduction of variational model . . . . . . . . . . . . . . . . . . . 4
1.4 Organization of the thesis . . . . . . . . . . . . . . . . . . . . . . . . 6

Chapter 2 Rician noise removal with DC Algorithm 8

2.1 Introduction and background . . . . . . . . . . . . . . . . . . . . . 8
2.2 Related work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11
  2.2.1 MAP model for Rician noise removal . . . . . . . . . . . . . . 11
  2.2.2 A convex MAP model . . . . . . . . . . . . . . . . . . . . . . . 12
  2.2.3 Another convex model for Rician noise removal . . . . . . . . 13
2.3 DC Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
2.4 DCA for Rician noise removal . . . . . . . . . . . . . . . . . . . . . 16
  2.4.1 Image denoising with DCA : model description . . . . . . . . . 16
  2.4.2 Convergence analysis of DC algorithm for MAP model . . . . 17
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>PSNR values and CPU-time (in second) for Rician noise removal.</td>
<td>26</td>
</tr>
<tr>
<td>2.2</td>
<td>PSNR values by different methods for Rician noise removal under blurry condition.</td>
<td>30</td>
</tr>
<tr>
<td>3.1</td>
<td>PSNR value of Cauchy noise removal.</td>
<td>39</td>
</tr>
<tr>
<td>3.2</td>
<td>PSNR value of Cauchy noise removal.</td>
<td>42</td>
</tr>
<tr>
<td>3.3</td>
<td>PSNR value of Cauchy noise removal under Gaussian blur.</td>
<td>42</td>
</tr>
<tr>
<td>4.1</td>
<td>Noise level estimation with SQ-Index</td>
<td>68</td>
</tr>
<tr>
<td>4.2</td>
<td>Performance of image denoising and deblurring under Gaussian blur (ISNR).</td>
<td>72</td>
</tr>
<tr>
<td>4.3</td>
<td>Performance of image denoising and deblurring under average blur (ISNR).</td>
<td>72</td>
</tr>
<tr>
<td>5.1</td>
<td>Performance of Gaussian noise removal with parameter selected by SQ-Index</td>
<td>75</td>
</tr>
<tr>
<td>5.2</td>
<td>Performance of Gaussian noise removal under blurry condition with parameter selected by SQ-Index</td>
<td>79</td>
</tr>
<tr>
<td>5.3</td>
<td>Performance of Poisson noise removal with parameter selected by SQ-Index</td>
<td>84</td>
</tr>
<tr>
<td>5.4</td>
<td>Performance of Poisson noise removal under Gaussian blur with parameter selected by SQ-Index</td>
<td>87</td>
</tr>
<tr>
<td>5.5</td>
<td>Performance of Rician noise removal with parameter selected by SQ-Index</td>
<td>89</td>
</tr>
<tr>
<td>5.6</td>
<td>Performance of Rician noise removal under blurry condition with parameter selected by SQ-Index</td>
<td>93</td>
</tr>
</tbody>
</table>
List of Figures

1.1 (a) Cameraman with Gaussian noise; (b) Cameraman with Poisson noise;
        (c) Cameraman with Rician noise; (d)-(f) noise residual of (a)-(c): . . . 2

2.1 Rician distribution with different SNR ratio. . . . . . . . . . . . . . . 9
2.2 Original images. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
2.3 Image corrupted by Rician noise. . . . . . . . . . . . . . . . . . . . . 24
2.4 Row 1: Image Cameraman restored with Rician noise $\sigma = 20$. Row 2:
        Zoom part of image restored by three methods for comparison. . . . 25
2.5 Row 1: PeMRI restored under Rician noise with $\sigma = 20$ by three
        methods. Row 2: Zoom part of image restored by three methods for
        comparison. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
2.6 image degraded with blur. . . . . . . . . . . . . . . . . . . . . . . . . 27
2.7 PSNR values of different methods when deblurring Gaussian blur with
        Rician noise $\sigma = 15$ for Brain and their residuals of different methods
        after restoration. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
2.8 PSNR values of different methods when deblurring motion blur with
        Rician noise $\sigma = 15$ for Cameraman and residuals of different methods
        after restoration. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
2.9 PSNR values of different methods when deblurring motion blur with
        Rician noise $\sigma = 15$ for image 'peMRI' and the residuals of different
        methods after restoration. . . . . . . . . . . . . . . . . . . . . . . . . . 29

3.1 Cauchy noise removal for $\xi = 0.02$. . . . . . . . . . . . . . . . . . . . 40
3.2 Cauchy noise removal for $\xi = 0.04$. . . . . . . . . . . . . . . . . . . . 41
3.3 Cauchy noise removal under Gaussian blur for variance 1 with $\xi = 0.02$. 43

4.1 Phase information can preserve edge. . . . . . . . . . . . . . . . . . . . 48
4.2 PSNR value of restored image with parameters selected with different \( q \) by SQ-Index for image with Gaussian blur. .................................................. 66

4.3 PSNR value of restored image with parameter selected with different \( q \) by SQ-Index for image with motion blur. .................................................. 67

4.4 SQ Index curve for constrained TV model for image (a) Parrot with 40 dB Gaussian noise; (b) Cameraman with 20 dB Gaussian noise. ....................... 69

4.5 (a) Cameraman with noise level 10; (b) DP method: ISNR=65.8dB; (c) SQ-Index: ISNR=65.8dB. ................................................................. 70

4.6 (a) Boat with noise level 20; (b) DP method: ISNR=65.1dB; (c) SQ-Index: ISNR=65.1dB. ................................................................. 70

5.1 Column 1: image corrupted with Gaussian noise with noise intensity 20; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: the sensitivity of \( q \) with regard to PSNR of restored image. .................................................. 76

5.2 Column 1: image corrupted with Gaussian noise with noise intensity 30; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: the sensitivity of \( q \) with regard to PSNR of restored image. .................................................. 77

5.3 Column 1: image corrupted with Gaussian noise with noise intensity 10 under Gaussian blur with size 9 * 9 and variance 3; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; column 4: sensitivity of \( q \) for SQ-Index with respect to PSNR of restored image. 80
5.4 Column 1: image corrupted with Gaussian noise with noise intensity 10 under average blur with size 9; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: sensitivity of $q$ for SQ-Index with respect to PSNR of restored image. 81

5.5 Column 1: image restored with best PSNR-value (try and trial) which is corrupted by Poisson noise with noise intensity 8; Column 2: image restored with model parameter selected by SQ-Index for $q = 1$; Column 3: image restored with model parameter selected by SQ-Index for $q = 1.15$; Column 4: sensitivity of $q$ with respect to PSNR of restored image. 85

5.6 Column 1: image restored with best PSNR-value (try and trial) which is corrupted by Gaussian blur with variance 1 and Poisson noise with noise intensity 8; Column 2: image restored with model parameter selected by SQ-Index; Column 3: sensitivity of $q$ with respect to PSNR of restored image. 88

5.7 Column 1: image corrupted with Rician noise with noise intensity 20; Column 2: image restored with model parameter selected by best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index. 90

5.8 Column 1: image corrupted with Rician noise with noise intensity 15 under Gaussian blur with variance 1; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index. 91

5.9 Column 1: image corrupted with Rician noise with noise intensity 15 under motion blur; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index. 92
Chapter 1

Introduction

Image processing has been widely used in different areas. It can be used for processing photos taken by people’s cameras, astronomy radio, radar imaging, medical devices and tomography. Photos may be corrupted with noise or blurred by translation, shake, out-of-focus, etc. It can also be corrupted by surrounding environment like reflected light. Image processing generally can include geometric transformations like enlargement, rotation, reduction, interpolation, image editing, image classification, super-resolution [91], feature extraction [54], image enhancement and image restoration. Image restoration is to restore the image to its original form or to detect the important information we want. Image restoration includes but not limited to image denoising, deblurring, inpainting and segmentation. In our thesis, we will deal with image denoising and deblurring by variational models. A popular theory called difference of convex algorithm (DCA) is adopted for solving non-convex models. Experimental result shows that a better local optimiser can be found by DCA comparing with by other methods. We also do research on image quality assessment. We propose a new sharpness metric called Sharpness q-Index (SQ-Index) to assess image quality, estimate noise level and help choose regularization parameters. In this chapter, we give a basic introduction of image denoising and deblurring and a overview of previous work for solving these problems.
1.1 Introduction of image denoising

Image denoising is a significant subject in processing of image since a real image obtained may be corrupted by some random noise. It can be classified by three types of noise: additive noise, multiplicative noise and impulse noise. Then different models featured by different properties of noise probability distribution are proposed for solving corresponded noise. Noise can also be divided by signal-dependant noise and signal-independent noise. Figure 1.1 shows image corrupted by different noise and what different noise is like. We observe that Gaussian noise is signal independent since the residual image has none of original image information. However, both Poisson and Rician noise are signal dependent, since the residual image contains some information of the original image. Many methods for solving image denoising problems are proposed during the last decades. Interested readers can refer to [15] for a good review. Some researchers use nonlinear partial differential equations (PDEs) [5,27,64,87] for image denoising while others use regularization techniques [18–20,63,66]. Some have
melted the two approaches together like [79]. For regularization method, the variational models [18–20, 63, 66] are very popular recently by dealing with image pixel in spatial space. Variational technique can also be used in color image, see [4]. The other regularization methods are conducted in frequency space, like wavelet shrinkage method [23, 55, 56] which is based on thresholds of wavelet coefficients of noisy image. By transforming the data into the wavelet base, the noise is removed by modifying these wavelet coefficients since the larger coefficients represents mainly the signal and the smaller ones represents mainly the noise. An alternative class of method is dictionary-learning method which uses redundant representations and sparsity as driving forces for denoising of signals. A general introduction is in [28]. Some also use a combination of sparse representation and variational method [74] for image denoising. All different types of denoising method is related and can be deducted from one to the others. The relations of different types of methods are introduced in [22].

1.2 Introduction of image deblurring

Besides noise, blur is another common form of image degradation. It appears from a variety of possible sources like camera defocus, optical aberrations of the lens, atmospheric scatter, etc. Image deblurring tries to recover the original image by solving a mathematical model of the blurring process. A overview of different blurs and methods for deblurring is introduced in [83]. There are different kinds of blur, including but not limited to Gaussian blur, motion blur, uniform blur. The blurred image is generated by the convolution of the original clean image with blur kernels. Figure 1.2 shows two typical blur kernels. Image deblurring methods can be divided into blind-deblurring and non-blind deblurring. Non blind deblurring is that we suppose we know the blur kernel or we can estimate the blur kernel before deblurring. The blind deblurring is that we can deblur the image in the same time to estimate the blur kernel by mathematical models. Many wavelets related methods like [16, 17, 71]
have been developed for blind deblurring.

1.3 Introduction of variational model

Image denoising and deblurring problem can be modelled as an inverse problem. For image with additive noise, we have:

\[ f = Au + n. \]  

(1.1)

where \( u \) is the clean image, \( A \) is a linear operator for image \( u \). \( A \) is an identity matrix for image denoising while for deblurring problems, \( A \) is a linear, continuous blur kernel and \( n \) is the additive noise. Solving \( u \) from \( f \) is mostly an ill-posed problem. Then it is very natural and necessary to use regularization technique which uses the prior knowledge of image to reconstruct the model for a reliable solution. The classical ROF model [66] removing Gaussian noise is:

\[ \hat{u} = \arg \min_{u \in BV(\Omega)} p(u) + \frac{\lambda}{2} \| Au - f \|_2^2, \]  

(1.2)

where \( \Omega \subset \mathbb{R}^N \) is an open bounded set with Lipschitz boundary. \( BV(\Omega) \) is the subspace of functions \( u \in L^1(\Omega) \) such that \( J(u) \) is finite. \( BV \) space is defined by \( BV(\Omega) = \{ u \in L^1(\Omega), \int_\Omega |Du| < \infty \} \). In other words, if \( u \) belongs to \( BV(\Omega) \), then it belongs to \( L^1(\Omega) \) and has bounded variation \( \int_\Omega |Du| < \infty \). For further details about the notation, please refer to [30,80]. The model is based on a Bayesian approach:

\[ P(u|f)P(f) = P(f|u)P(u). \]  

(1.3)
For the inverse problem \( f = Au + n \), we consider a priori probability density for original signals \( P(u) : e^{-P(u)}du \) and a posteriori probability for \( u \) knowing \( f \). The density for the probability of \( f \) knowing \( u \) is equivalent with the density of \( n = f - Au \). When \( n \) is Gaussian noise defined above. Then the density of Gaussian noise \( n \) is:

\[
e^{-\frac{1}{2\sigma^2}\|f-Au\|^2}.
\] (1.4)

By (1.3) the probability \( P(u|f) \) is:

\[
P(u|f) = \frac{P(f|u)P(u)}{P(f)} = \frac{1}{Z(f)}e^{-p(u)}e^{-\frac{1}{2\sigma^2}\|f-Au\|^2}.
\]

\( Z(f) \) is a re-normalization factor:

\[
Z(f) = \int e^{-p(u)}e^{-\frac{1}{2\sigma^2}\|f-Au\|^2} du.
\]

Maximum a posteriori (MAP) image reconstruction aims at maximizing this probability \( P(u|f) \), or solving the minimization problem:

\[
\min_u p(u) + \frac{1}{2\sigma^2}\|f - Au\|^2.
\] (1.5)

Now we consider the minimization problem (1.5) in continuous setting:

\[
\min_{u \in L^2(\Omega)} \lambda p(u) + \frac{1}{2} \int_\Omega \|f - Au\|^2 dx,
\] (1.6)

where \( p(u) = \int_\Omega |\nabla u(x)| dx \) represents a priori probability density \( p(u) \) which is well defined for \( C^1 \) functions. More importantly, \( p(u) \) is a convex function and many convex optimization methods can be used to solve it. We give the two propositions of \( p(u) \) below.

**Proposition 1.** \( p(u) \) is convex. In other word, \( \forall u_1, u_2 \) and \( t \in [0, 1] \), we have:

\[
p(tu_1 + (1-t)u_2) \leq tp(u_1) + (1-t)p(u_2).
\] (1.7)

The proof is obvious since \( p(u) \) is the norm of linear function. By the triangle inequality of the norm property, we get the proof.

**Proposition 2.** \( p(u) \) is positively homogeneous. For each \( u \) and \( t > 0 \),

\[
p(tu) = tp(u).
\]
1.4 Organization of the thesis

The organization of this thesis is as follows. In Chapter 2, we apply a numerical algorithm called difference of convex algorithm (DCA) to solve non-convex variational models. We apply DCA to non-convex Rician noise removal MAP model. The objective is to find an alternative method to solve non-convex model for image restoration. Compared with gradient descent method, our method can find a better solution and the computing time is largely reduced. The performance is also very competitive with other convex models.

In Chapter 3, we apply DCA to solve non-convex variational model for Cauchy noise removal. The objective is to find a method to solve directly non-convex model for image restoration. Compared with the convex model, the performance of our method is very competitive.

In Chapter 4, we introduce a new sharpness metric: SQ-Index. Based on the theory of Global Phase Coherence, phase information reserves the contours of image. We then give the details about the deduction of SQ-Index based on existed S-Index. We apply SQ-Index to regularization parameter selection. We show examples of how SQ-Index can be used to choose regularization parameter for a constrained TV model. Here we also contribute to create a new primal-dual projection algorithm to solve directly the constrained TV-model. We show performance of image denoising and deblurring under Gaussian noise using SQ-Index to select the parameters in the model. Comparison with other parameter selection methods is also given. On average, our method has the best performance for various imaging problems.

In Chapter 5, to show the effectiveness and robustness of SQ-Index for parameter selection, an extensive experiments are made. We apply SQ-Index to TV denoising model, TV deblurring and denoising model, Poison denoising model, Poisson denoising and deblurring model, Rician denoising MAP model, Rician deblurring and denoising model. We show that SQ-Index can choose well the regularization parameters for all above models. With selected parameters, the variational models can
restore image efficiently with the PSNR which is close to the best possible PSNR value that we can recover manually.

Chapter 6 concludes our thesis and discusses our future work.
Chapter 2

Rician noise removal with DC Algorithm

2.1 Introduction and background

The magnetic resonance image (MRI) is a medical imaging technique used in hospitals and clinics for medical diagnosis. It has many applications like CT, MRI scanning. During the generation of MR images, there always appears noise and blur which will largely affect the judgement of diagnosis. Usually, the real and imaginary parts of the MR image are both corrupted by two independent random Gaussian white noise which follows the same distribution. A famous introduction of Rician noise is in [37]. Transform the noise to magnitude MR images, we can see the noise follows a Rice distribution or Rician distribution. Images with Rician noise can be composed mathematically as:

\[ f = f_R + f_I \]
\[ = u + \eta_1 + \eta_2 i, \]

where \( u \) is the true amplitude of the image and \( \eta_1, \eta_2 \) are two independent Gaussian noise with distribution \( N(0, \sigma^2) \). In this case, the magnitude image \( f \) is:

\[ f = \sqrt{(u + \eta_1)^2 + \eta_2^2}. \]

Suppose image with presence of Rician noise is noted \( f \), image without the presence of Rician noise is noted \( u \). Based on statistical lemma in [42], the probability distribution
for $f$ is:

$$P(f) = \frac{f}{\sigma^2} e^{\frac{u^2}{2\sigma^2}} I_0\left(\frac{uf}{\sigma^2}\right).$$  \hspace{1cm} (2.1)$$

where $I_0$ represents the modified zero order Bessel function of the first kind and $\sigma$ is the standard deviation of the Gaussian noise in the real and the imaginary part of images. Rician noise is signal dependant and thus can generate random fluctuation and brings a bias to the original image which will reduce the image contrast. It is more difficult to remove Rician noise compared with additive noise like Gaussian noise since it is signal dependent. In high SNR (signal noise ratio) case, Rician noise can be approximated by Gaussian noise, however, when in low SNR case (SNR < 2), Rician noise is quite different with Gaussian noise and it is necessary to develop new techniques to remove Rician noise. Figure 2.1 shows the histogram of Rician distribution with different SNR ratios.

Since Rician noise follows a Rician distribution, it appears a $I_0$ which is a Bessel function. It is the solution of the equation:

$$x^2 \frac{d^2 x}{dy^2} + x \frac{dy}{dx} + (x^2 - \alpha^2)y = 0.$$  \hspace{1cm} (2.2)$$

Integral forms of modified Bessel functions of the first kind with integer orders $n$ are:

$$I_n(x) = \frac{1}{\pi} \int_0^\pi \cos(n\theta) \exp(x \cos \theta) d\theta.$$  \hspace{1cm} (2.3)$$

Then we can easily get:

$$I'_0(x) = I_1(x),$$

$$I'_1(x) = \frac{1}{2}(I_0(x) + I_2(x)).$$
Lemma 1. \( \forall x \in \mathbb{R}, \frac{I_1(x)}{I_0(x)} \) is finite.

Proof. From the definition of \( I_0(x), I_1(x) \), we have:

\[
I_0(x) = \frac{1}{\pi} \int_{0}^{\pi} \exp(x)d\theta,
\]

\[
I_1(x) = \frac{1}{\pi} \int_{0}^{\pi} \cos(n\theta) \exp(x \cos \theta)d\theta.
\]

Since \( \cos(n\theta) \in [-1, 1] \), then we have:

\[
I_1(x) = \frac{1}{\pi} \int_{0}^{\pi} \cos(n\theta) \exp(x \cos \theta)d\theta \leq \frac{1}{\pi} \int_{0}^{\pi} \exp(x)d\theta = I_0(x)
\]

\[
I_1(x) = \frac{1}{\pi} \int_{0}^{\pi} \cos(n\theta) \exp(x \cos \theta)d\theta \geq -\frac{1}{\pi} \int_{0}^{\pi} \exp(x)d\theta = -I_0(x)
\]

Then we have:

\[
-1 \leq \frac{I_1(x)}{I_0(x)} \leq 1
\]

\[
\square
\]

Many methods have been proposed to restore images corrupted by both blur and Rician noise [2, 6, 7, 26, 60, 89]. In [60], Nowak studied wavelet method for Rician noise estimation and removal. Basu, Fletcher and Whitaker [7] applied anisotropic diffusion in diffusion tensor MRI data, using a correction term to remove Rician noise. Descoteaux and Wiest-Daessle et al [26] applied a non-local means filter to remove Rician noise. Wang and Zhou [84] performed MRI denoising with a combination of TV and wavelet based regularization. Getreuer developed a MAP model [34] based on maximum a posteriori estimation property which is very efficient for Rician noise removal. The MAP model consists of a total variation (TV) regularization with a fidelity term involving the Rician probability distribution. The model was solved by classical \( L^2 \) gradient descent and Sobolev gradient descent. Since the model is not convex, the above numerical methods may converge to some local minimizers, and different initialization values may give different solutions. In order to solve the instability of the solution from non-convex model, he also proposes a convex approximated
model to the MAP model which can ensure a global minimum solution not depending on the initiation condition. However, this method has a very complicated explicit form for the model and is hard to understand. Since the form of the model is approximated, the solution also exists a bias to the theoretical optimal solution. Then, Chen and Zeng [25] proposed another convex model based on basic MAP model by adding another regularization item. The disadvantage of this model is that it exists a bias and need an extra step for bias correction. In order to solve the non-convex model directly and more efficiently, we apply another popular numerical technique: DC Algorithm (DCA). We show that DCA method can get a better local optimizer. In the following sections, we show how to remove Rician noise with DCA method. First, we give a brief introduction of DCA and its properties. Then we give the proof of the solution’s convergence and the algorithm for solving MAP model with DCA for Rician noise removal. The comparison with other methods is shown in the experimental section. Our DCA method gives a better performance. Experiment result shows that our performance outperforms the gradient descent method and is very competitive compared with the two convex models.

2.2 Related work

2.2.1 MAP model for Rician noise removal

Now we introduce the MAP model for Rician noise removal. The MAP model proposed by [34] for Rician noise removal is based on Bayesian approach:

\[
\hat{u} = \arg \max P(u|f) = \arg \max \frac{P(f|u)P(u)}{P(f)} = \arg \min -\log(P(f|u)) - \log(P(u)).
\]

Based on (2.1), we have:

\[
P(f|u) = \frac{f}{\sigma^2} e^{-\frac{f^2 + u^2}{2\sigma^2}} I_0\left(\frac{uf}{\sigma^2}\right),
\]
Then,

\[-\log(P(f|u)) = -\int_{\Omega} \log(P(f(x)|u(x)))dx\]

\[= \frac{1}{2\sigma^2} \int_{\Omega} (u(x)^2 + f(x)^2)dx - \int_{\Omega} \log I_0(f(x)u(x)) - \int_{\Omega} \log(f(x))dx.\]

Based on Gibbs prior [3] and total variation approach,

\[P(u) = \exp(-\gamma p(u)) = \exp(-\gamma \int_{\Omega} |\nabla u(x)|dx).\]

Thus, we get the MAP model proposed by [34]:

\[\hat{u} = \arg \inf_{u} \frac{1}{2\sigma^2} \int_{\Omega} u^2dx - \int_{\Omega} \log I_0(fu) + \gamma \int_{\Omega} |\nabla u|dx. \quad (2.4)\]

### 2.2.2 A convex MAP model

In [34], Getreuer also proposed an improved approximated convex model which permits the split bregman numerical algorithm [35] to solve this problem. The model is written as:

\[\inf_{u} \lambda \int_{\Omega} H_\sigma(z, f) + \int_{\Omega} |Du|dx, \quad (2.5)\]

where \(z = Au\) and

\[H_\sigma(z) = \frac{f^2 + z^2}{2\sigma^2} - \log I_0(f \frac{z}{\sigma^2}),\]

\[H'_\sigma(z, f) = \frac{z f}{\sigma^2} \frac{I_1(f \frac{z}{\sigma^2})}{I_0(f \frac{z}{\sigma^2})}.\]

Since \(\frac{I_1(t)}{I_0(t)} \approx \frac{t^3 + 0.950037t^2 + 2.38944t + 4.65314}{t^3 + 1.48937t^2 + 2.87541t + 4.65314} = A(t)\), then \(H'_\sigma(z)\) can be approximated as \(H'_\sigma(z) = \frac{z}{\sigma^2} - \frac{f}{\sigma^2} A(f \frac{z}{\sigma^2})\). Then Getreuer proves that we can make a convex approximation of \(H_\sigma\) as:

\[G_\sigma(u, f) = \begin{cases} 
H_\sigma(z) & \text{if } z \geq c \sigma, \\
H_\sigma(c \sigma) + H'_\sigma(c \sigma)(z - c \sigma) & \text{if } z \leq c \sigma,
\end{cases}\]

with \(c \leq 0.8246\).
2.2.3 Another convex model for Rician noise removal

Based on MAP model proposed by Getreuer [34], Chen and Zeng [25] proposed another convex model for Rician noise removal by adding another regularization term to the original MAP model. The new convex model is:

$$\hat{u} = \arg \inf_u \frac{1}{2\sigma^2} \int_{\Omega} u^2 dx - \int_{\Omega} \log I_0(\frac{fu}{\sigma^2}) dx + \frac{1}{\sigma} \int_{\Omega} (\sqrt{u} - \sqrt{f})^2 dx + \gamma \int_{\Omega} |\nabla u| dx. \quad (2.6)$$

Chen and Zeng already proved the convexity of this model under mild conditions thus primal-dual method can be used to get the solution.

2.3 DC Algorithm

In image processing, we always rely on the use of convex optimization to ensure that solutions exist and could be computed. However, for many cases, we need to deal with models which are non-convex. Many popular numerical algorithms based on convex functions are not efficient for solving non-convex models. For the moment, there are few methods for solving non-convex models. In this chapter, we want to apply the theory of optimization for a superclass of convex functions, called difference of convex algorithm for solving non-convex model for image restoration. Pham-Dihn created DC theory and he extended his theory in [41, 76]. Other researchers also do a lot of research for the development of DC theory. We can refer to [67, 68, 90]. In this section, we want to introduce the basic DC theory and with the properties of DC, we can apply it to solve our MAP model for Rician noise removal. Now we give a brief review of the basic DCA.

Function $F \in \Gamma_0(\mathbb{R}^n)$ where $\Gamma_0(\mathbb{R}^n) := \{ f : \mathbb{R}^n \to \mathbb{R} \cup \{ +\infty \} \}$. To solve:

$$\inf F(x) : x \in \mathbb{R}^n. \quad (2.7)$$

We divide $F$ by a difference between function $G$ and function $H$, where $G$ and $H$ are lower semi-continuous proper convex functions, such that:

$$F(x) = G(x) - H(x) \quad \forall x \in \mathbb{R}^n.$$
Unlike the sum, the difference of convex functions destroys convexity. The key idea of DCA is an extension of convex analysis to non smooth, non convex analysis based on convex programming. The primal problem is:

\[ \alpha_p = \inf_{u \in \mathbb{R}^n} F(u) = \inf_{u \in \mathbb{R}^n} G(u) - H(u). \]  

(2.8)

DC program has a perfect duality property. Before we give the dual form, let us first give the definition of a conjugate function.

**Definition 1.** Let \( g \in \Gamma_0(\mathbb{R}^n) \), the conjugate function \( g^* \) is defined by:

\[ g^*(y) := \sup\{ \langle x, y \rangle - g(x) : x \in \mathbb{R}^n \}. \]  

(2.9)

We can deduce that \( g^{**} = g \).

Now we have the dual form of (2.8) as:

\[ \alpha_d = \inf_{v \in \mathbb{R}^n} H^*(v) - G^*(v), \]  

(2.10)

where \( G^*, H^* \) denote their dual functions of \( G, H \) respectively.

Now we summarize the properties of DCA which can be used in the next section.

**Proposition 3.** For primal DC program (for dual DC program by similarity), \( x^* \) is called a critical point of \( G - H \) or generalized KKT point for (2.8) if \( \partial H(x^*) \cap \partial G(x^*) \neq \emptyset \) and \( \emptyset \neq \partial H(x^*) \subset \partial G(x^*) \).

Pham Dinh Tao has proved that for non-convex \( F \), the critical point \( x^* \) is a local minimum. If \( F \) is convex, then \( x^* \) is the global minimum.

Problem (2.8) and (2.10) are mutually dual since:

\[
\alpha_p = \inf_{x \in X} G(x) - H(x)
= \inf \left\{ \inf_{x \in X} \{ G(x) - \sup_{y \in Y} \{ x^T y - H^*(y) \} : x \in X \} \right\}
= \inf \left\{ \inf_{x \in X} \{ G(x) - x^T y + H^*(y) \} : \right. 
= \inf \left\{ H^*(y) - G^*(y) : y \in Y \right\} 
= \alpha_d.
\]
We replace $H(u)$ in (2.8) by its affine minimization around $u^k$:

$$
H(u) = H(u^k) + \langle u - u^k, v^k \rangle, \quad v^k \in \partial H(u^k).
$$

(2.11)

Then the primal and dual problems become:

$$
\inf (G - H)_k(x) : = G(x) - H(x^k) + \langle x - x^k, y^k \rangle \forall x \in \mathbb{R}^n, 
$$

(2.12)

$$
\inf (H^* - G^*)_k(y) : = H^*(y) - G^*(y^k) + \langle y - y^k, x^{k+1} \rangle \forall y \in \mathbb{R}^n.
$$

(2.13)

There are infinite decomposition for DCA since $F(x) = G(x) - H(x)$ can be also $F(x) = G(x) + e(x) - (H(x) + e(x))$. Thus, how to choose a good pair of $G(x)$ and $H(x)$ is also a technique. Researches have been conducted in [77, 78].

Since DC function $F$ has infinitely many DC decompositions which may result in different qualities of image restoration by the different speed of convergence, robustness. In practice, we try to choose $G$ and $H$ such that the numerical algorithm can be easily solved either they are in an explicit form or the calculation is fast. Convergence property of DCA is discussed in [76]. If $x^k$ is bounded, limit points of $x^k$ is finite and $\lim_{k \to +\infty} \|x^{k+1} - x^k\| = 0$, then DCA solution is convergent. The general DC algorithm can be described as below:

- Initiate $x^0 \in \mathbb{R}^n$. Set $k = 0$. Do until $\frac{\|x^{k+1} - x^k\|}{\|x^k\|} \leq \epsilon$.

- Calculate $y^k$ by $y^k = \partial H(x^k)$.

- Calculate $x^{k+1}$ by $x^{k+1} \in \arg \min \{G(x) - [H(x^k) + \langle x - x^k, y^k \rangle] : x \in \mathbb{R}^n\}$.

- $k = k + 1$. 

2.4 DCA for Rician noise removal

2.4.1 Image denoising with DCA: model description

Now we use DCA to solve MAP model for Rician noise removal. We give the MAP model in discrete form:

\[ \min_{u \in \Omega} F(u) = \min_{u \in \Omega} \frac{1}{2 \sigma^2} \|u\|_2^2 - \langle \log(I_0(\frac{f \cdot u}{\sigma^2}), 1) + \gamma \text{TV}(u). \tag{2.14} \]

where TV(u) := \|\nabla u\|_1 = \sqrt{\|\nabla_x u\|^2 + \|\nabla_y u\|^2}. The \|\nabla_x u\| is defined as \( u_{i+1,j} - u_{i,j} \) and the \|\nabla_y u\| is defined as \( u_{i,j+1} - u_{i,j} \). The dual form of TV can be shown as:

\[ \text{TV}(u) = \max_{p \in P} \langle \nabla u, p \rangle, \text{ where } p \text{ is in the set: } \{p||p|| \leq 1 \}. \]

Please see [20] for reference.

In order to use DCA, we separate \( F(u) \) to \( G(u) \) and \( H(u) \) naturally as:

\[ G(u) = \frac{1}{2 \sigma^2} \|u\|_2^2 + \gamma \text{TV}(u), \tag{2.15} \]
\[ H(u) = \langle \log(I_0(\frac{f \cdot u}{\sigma^2}), 1) \rangle. \tag{2.16} \]

Lemma 2. \( G(u), H(u) \) in (2.15), (2.16) are convex functions respectively.

Proof.

For \( G(u) \), it is very obvious that \( G(u) \) is convex since every norm is convex by the triangle inequality and positive homogeneity.

For \( H(u) \), the proof is given in [25], Appendix 3.

From the general algorithm mentioned in the previous section, we have:

\[ v^k = \partial H(u^k) = \frac{f}{\sigma^2} I_0(\frac{f}{\sigma^2}); \tag{2.17} \]
\[ u^k = G(u) - \langle v_k, u \rangle, \]
\[ = \arg \min_{u \in \Omega} \frac{1}{2 \sigma^2} \|u\|_2^2 + \gamma \text{TV}(u) - \langle v^k, u \rangle. \tag{2.18} \]

By dual transform of TV norm, \( u^k \) in (2.18) is equivalent with:

\[ u^k = \arg \min_{u} \min_{p} \frac{1}{2 \sigma^2} \|u\|_2^2 - \gamma \langle u, \text{div } p \rangle - \langle v^k, u \rangle, \tag{2.19} \]
where $\|p\| \leq 1$.

The Lagrangian for (2.19) is:

$$L(u,p) = \frac{1}{2\sigma^2} \|u\|^2 + \gamma \langle u, -\text{div}p \rangle - \langle v^k, u \rangle.$$  \hfill (2.20)

Therefore, (2.19) can be formulated as a saddle point problem:

$$\min_{u \in \Omega} \max_{p \in P} \{ \hat{E}(u, p) \},$$ \hfill (2.21)

and

$$\hat{E}(u, p) = \frac{1}{2\sigma^2} \|u\|^2 + \gamma \langle u, -\text{div}p \rangle - \langle v^k, u \rangle.$$ \hfill (2.22)

Then the problem (2.22) can be separated as two sub-problems as:

$$p^{k+1} = \arg \max_{p \in P} \gamma \langle \nabla u^k, p \rangle - \frac{1}{2\beta} \|p - p^k\|^2,$$ \hfill (2.23)

$$u^{k+1} = \arg \min_{u} \frac{1}{2\sigma^2} \|u\|^2 - \gamma \langle \text{div}p^{k+1}, u \rangle - \langle v_k, u \rangle + \frac{1}{2\tau} \|u - u^k\|^2.$$ \hfill (2.24)

The solution of (2.23) and (2.24) are respectively:

$$p^{k+1} = p^k + \beta \gamma \nabla u^k, \quad p^{k+1} = \frac{\hat{p}^{k+1}}{\max(1, \|\hat{p}^{k+1}\|)};$$ \hfill (2.25)

$$u^{k+1} = \frac{\gamma \tau \text{div}p^{k+1} + \tau v_k + u^k}{1 + \frac{\tau}{\sigma^2}}.$$ \hfill (2.26)

### 2.4.2 Convergence analysis of DC algorithm for MAP model

We now prove that the sequence $u^k$ got from (2.26) is convergent to a stationary point $u^*$. We reformulate our model as standard primal-dual model:

$$\min_{x} \max_{y} \langle y, Lx \rangle - J^*(y) + K(x),$$ \hfill (2.27)

where $x = u, y = p, L = \nabla, J^* = \chi^*$ where $\chi^*$ is the characteristic function of the closed convex set, see [18]. And $K(x) = -\langle v^k, x \rangle + \frac{1}{2\sigma^2} \|x\|^2$.

**Lemma 3.** The set of saddle points of problem (2.27) is non empty.
Proof. From [40] (p333, 334, theorem 4.3.1), if the conditions \((H1) - (H4)\) in [40] are satisfied, then the saddle points set is non empty. It is very straight-forward to prove that our model satisfies the above conditions. Then the saddle point of problem (2.27) exists. \(\square\)

Lemma 4. Assume that \(u^k\) is the sequence generated from DCA algorithm for solving (2.19), then we have the following inequality:

\[
F(u^k) - F(u^{k+1}) \geq 0.
\]

Proof. \(v^k \in \partial H(u^k)\), from the definition of sub-differential, we have:

\[
H(u^{k+1}) \geq H(u^k) + \langle u^{k+1} - u^k, v^k \rangle.
\]

(2.28)

If \(u^k\) is the solution of problem (2.8), then \(v^k \in \partial G(u^{k+1})\). Then we also have:

\[
G(u^k) \geq G(u^{k+1}) + \langle u^k - u^{k+1}, v^k \rangle.
\]

(2.29)

Adding (2.28) and (2.29), we have:

\[
F(u^k) - F(u^{k+1}) \geq 0.
\]

\(\square\)

According to Lemma 3, the saddle point exists. Then we set \(u^*\) be the solution of (2.27). For any \(k\), \(F(u^k) \geq F(u^*)\). Then we prove that \(F(u^k)\) is bounded below. From Lemma 4, \(F(u^k)\) is decreasing monotonically and bounded from below, thus convergent. Then we get that \(|u^{k+1} - u^k| \rightarrow 0\) for a weak convergence.

Theorem 1. Any limit point \(u^*\) of \(u^n\) satisfies a weak first-order optimality condition:

\[
0 \in \frac{u^*}{\sigma^2} + \gamma \partial |Du^*| - \frac{I_1(f_u^*)}{I_0(f_u^*)} \cdot \frac{f}{\sigma^2}.
\]

(2.30)
Proof. Based on the Bolzano-Weierstrass theorem, since \( u^k \) is bounded (0 \( \leq u^k \leq 255 \) for instance), then there exists a subsequence of \( u^k \) noted as \( u^{n_k} \), converging to a limit point \( u^* \). The optimality condition at the \( n^k \) step of the DC algorithm is:

\[
0 \in \frac{u^{n_k}}{\sigma^2} + \gamma \partial |D u^{n_k}| - \left( \frac{I_1(u^{n_k})}{I_0(u^{n_k})} \right) \cdot \frac{f}{\sigma^2}. \tag{2.31}
\]

As \( u^{n_k} \to u^* \), we have \( Du^{n_k} \to Du^* \), since \( |D| < \infty \). we have also \( \frac{u^{n_k}}{\sigma^2} \to \frac{u^*}{\sigma^2} \).

Moreover, the function \( l(z) = \frac{l_1(z)}{I_0(z)} \) is in a closed set (-1,1) the finite as we proved in the previous chapter. Then we have: \( \frac{l_1(u^{n_k})}{I_0(u^{n_k})} \to \frac{l_1(u^*)}{I_0(u^*)} \). Combining all the items, we arrive at (2.30).

Since \( u^* \) satisfies the optimality condition, the saddle point \( (u^k, p^k) \) exists. Algorithm 1 illustrates how to use DCA for Rician noise removal.

Algorithm 1 Algorithm for Rician noise removal

1: Fixed \( \gamma, \tau, \beta, maxOutIterNum, maxInIterNum, tol_{In}, tol_{Out} \). Initiate: \( u_0 = f, p = (0, 0, \cdots, 0) \in \mathbb{R}^3 \)

2: for \( k = 1 \) to \( maxOutIterNum \) do

3: \( v^k = \frac{l_1(u^k)}{I_0(u^k)} \cdot \frac{f}{\sigma^2} \).

4: for \( m = 1 \) to \( maxInIterNum \) do

5: calculate \( p^{m+1} \) by (2.25).

6: calculate \( u^{m+1} \) by (2.26).

7: if \( \frac{\|u_m - u_{m+1}\|_2}{\|u_m\|_2} < tol_{In}, u^k = u^{m+1}, \text{break; } \)

8: end for

9: if \( \frac{\|F(u_m) - F(u_{m+1})\|_2}{\|F(u_m)\|_2} < tol_{Out}, \text{break; } \)

10: end for

In order to prove the primal-dual algorithm is convergent, let us give the following theorem.

**Theorem 2.** Take \( \tau, \beta \), such that, \( \tau \beta \leq \frac{1}{8\gamma} \), then the sequence \( (u^k, p^k) \) defined in (2.26) and (2.25) converges to the saddle point \( (u^*, p^*) \) of (2.27).
Proof. Based on theorem 1 in [20], when $\tau_\beta \|L\|_2^2 < 1$, the numerical solution is convergent. Readers can refer to the proof in [20]. From [20], we know $|\nabla|^2 \leq 8$. Then we get the conclusion that if $\tau_\beta \leq \frac{1}{8\gamma^2}$, the numerical solution is convergent.

2.4.3 Image denoising and deblurring with DCA : model description

It is quite often that when a medical image is generated, noise and blur appear simultaneously. Thus, we also propose the discrete version of MAP model for Rician denoising under blurry condition:

$$
\min_{u \in \Omega} F(u) = \min_{u \in \Omega} \frac{1}{2\sigma^2} \|Au\|_2^2 - \langle \log(I_0(\frac{f \cdot Au}{\sigma^2}), 1) \rangle + \gamma TV(u),
$$

(2.32)

where $A$ is the blur kernel, $G(u)$ and $H(u)$ is defined naturally as:

$$
G(u) = \frac{1}{2\sigma^2} \|Au\|_2^2 + \gamma TV(u);
$$

(2.33)

$$
H(u) = \langle \log(I_0(\frac{f \cdot Au}{\sigma^2}), 1) \rangle.
$$

(2.34)

Then,

$$
v^k = \partial H(u^k) = A^T \frac{f}{\sigma^2} \frac{I_1(\frac{f \cdot Au}{\sigma^2})}{I_0(\frac{f \cdot Au}{\sigma^2})};
$$

(2.35)

$$
u^k = \arg \min_{u \in \Omega} \frac{1}{2\sigma^2} \|Au\|_2^2 + \gamma TV(u) - \langle v^k, u \rangle.
$$

(2.36)

Now we introduce another variable $w$, set $w = Au$, and the dual variable $q$ in addition, by dual transform of TV norm, $u^k$ in (2.36) is equivalent with:

$$
u^k = \arg \min_{u,w} \max_{p,q} \left\{ \hat{E}(u, w, p, q) \right\}
$$

(2.38)

where

$$
\hat{E}(u, w, p, q) = \frac{1}{2\sigma^2} \|w\|_2^2 - \gamma \langle u, \text{div} \ p \rangle - \langle v^k, u \rangle + \langle Au - w, q \rangle.
$$

(2.39)
We now prove that the sequence $u^k$ got from (2.36) is convergent to a stationary point $u^*$. We reformulate our model as standard primal-dual model:

$$
\min_{x} \max_{y} \langle y, Lx \rangle - J^*(y) + K(x),
$$

where $x = (u, w)^T$, $y = (p, q)^T$, $L = \begin{bmatrix} \gamma \nabla & 0 \\ A & -I \end{bmatrix}$, $J^* = (\chi^*, 0)$ where $\chi^*$ is the dual function of TV term $J(u) = |\nabla u|$, see [20] and $K = (-v^k, \frac{1}{2\sigma^2}||x||^2)$. The blur kernel is proved to be bounded [21], such that $||A||^2 \leq 1$.

Then we can similarly prove that the saddle point for problem (2.40) exists and the saddle point $x^* = (u^*, w^*), y^* = (p^*, q^*)$ satisfies the optimal condition:

$$
0 \in A^T Au^* + \gamma \partial |Du^*| - A^T \left( \frac{f_1(Au^*)}{f_0(\frac{A^2}{\sigma^2})} \cdot \frac{f}{\sigma^2} \right).
$$

(2.41)

Then for problem (2.40), we can separate the problem as four sub-problems:

$$
p^{m+1} = p^m + \beta \gamma \nabla u^m; \quad p^{m+1} = \frac{\bar{p}^{m+1}}{\max(1, |\bar{p}^{m+1}|)}; \quad (2.42)
$$

$$
q^{m+1} = \arg \max \langle Au^m - w^m, q \rangle - \frac{1}{2\beta} \|q - q^m\|^2; \quad (2.43)
$$

$$
w^{m+1} = \arg \min \frac{1}{2\sigma^2} \|w\|^2 - \langle w, q^{m+1} \rangle + \frac{1}{2\tau} \|w - w^m\|^2; \quad (2.44)
$$

$$
\bar{u}^{m+1} = \arg \min -\gamma (\text{div } p^{m+1}, u) + \langle Au, q^{m+1} \rangle - \langle v_k, u \rangle + \frac{1}{2\tau} \|u - u^m\|^2. \quad (2.45)
$$

The numerical solutions for problem (2.42), (2.43), (2.44), (2.45) are respectively:

$$
p^{m+1} = p^m + \beta \gamma \nabla u;
$$

$$
q^{m+1} = q^m + \beta (Au^m - w^m); \quad (2.47)
$$

$$
w^{m+1} = \frac{\tau q^{m+1} + w^m}{\sigma^2 + 1}; \quad (2.48)
$$

$$
u^{m+1} = u^m + \tau (\gamma \text{div } p^{m+1} + v^m - A^T q^{m+1}). \quad (2.49)
$$

We show our DCA algorithm for Rician denoising under blurry condition in Algorithm 2.4.3. In order to prove the primal-dual algorithm is convergent, let us give the following theorem.
Algorithm 2 Algorithm for image denoising and deblurring for Rician noise removal

1: Fixed $\gamma, \tau, \beta, maxOutIterNum, maxInIterNum, tol_{in}, tol_{out}$. Initiate: $u_0 = f, w_0 = f, p = 0, q = 0$.

2: for $k = 1$ to $maxOutIterNum$ do

3: \[ v^k = A^T f(\frac{A u^k f}{\lambda_0(g_0)} f) \]

4: for $m = 1$ to $maxInIterNum$ do

5: calculate $p^{m+1}$ by (2.46).

6: calculate $q^{m+1}$ by (2.47).

7: calculate $w^{m+1}$ by (2.48).

8: calculate $u^{m+1}$ by (2.49).

9: If $\frac{\|u_m - u_{m+1}\|}{\|u_m\|} < tol_{in}, u^k = u^{m+1}$, break;

10: end for

11: If $\frac{\|F(u_m) - F(u_{m+1})\|}{\|F(u_m)\|} < tol_{out}$, break;

12: end for

Theorem 3. Let $x^k = (u^k, w^k)^T, y^k = (p^k, q^k)^T$ which was defined in (2.46)-(2.49), we choose $\beta, \tau$, such that, $\tau \beta \leq \frac{1}{8\gamma^2+2}$, then the sequence $(x^k, y^k)$ converges to a saddle point $(x^*, y^*)$ of (2.40).

Proof. Based on theorem 1 in [20], when $\tau \beta \|L\| < 1$, the numerical solution is convergent. Readers can refer to the proof in [20]. Here $\|L\| = \gamma^2 \|\nabla\|^2 + \|A\|^2+1$. From [20], we know $\|\nabla\|^2 \leq 8$. From [21], we know that $\|A\|^2 \leq 1$. Then we get the conclusion that $\tau \beta \leq \frac{1}{8\gamma^2+2}$. \hfill \Box

2.5 Numerical results

In this section, we show the performance of DCA method for Rician noise removal, Rician noise and blur removal respectively. The images we test are in Figure 2.2 which includes gray-level images: Cameraman, medical MR images: brain, peMRI and MRRliver. The simulations are implemented in Matlab R2013a. We will compare
the PSNR [44] and computational time for those above methods. The performance is measured by PSNR (peak signal-to-noise ratio) which is defined as:

$$\text{PSNR} = 10 \log_{10} \left( \frac{\max(\hat{u})^2}{\frac{1}{NM} \left( \sum_{i=1}^{M} \sum_{j=1}^{N} (\hat{u}(i,j) - u(i,j))^2 \right)} \right),$$

where $\hat{u}$ is the original image of size $M$-by-$N$ and $u$ is the restored image.

### 2.5.1 Image denoising performance

In this part, we compare our DCA method with original MAP model with gradient descent method (MAP-GD) [34] and with Chen’s convex models [25] (LY model). The parameters are carefully selected for the three methods in the sense that we tried a series of parameters and chose the one which gave the best PSNR value. We test images with Rician noise corrupted by $\sigma = 20$ and $\sigma = 30$ respectively. The stopping criteria for three methods is set as:

$$\frac{\|u^{k+1} - u^k\|}{\|u^k\|} < \epsilon \quad \text{(2.50)}$$

where $\epsilon < 10^{-4}$.

For DCA method, we also need another stopping criteria for $F(u)$ as:

$$\frac{\|F(u^{k+1}) - F(u^k)\|}{\|u^k\|} < \epsilon \quad \text{(2.51)}$$

where $\epsilon < 10^{-4}$. Figure 2.3 shows the three images corrupted by Rician noise. Figure
2.4 and 2.5 show denoising performance for Cameraman, Brain and PeMRI respectively.

From Figure 2.4, we observe the zoomed image for the hand of cameraman and see that our DCA method can well restore the hand while the other two methods cannot.

We can see from the denoising PSNR that our DCA numerical method outperforms the other methods. Compared with MAP-GD method, we use exactly the same model, only numerical algorithms are different. We have a robust performance no matter the Rician noise is large or small. However, the MAP-GD method is inclined to have a relatively weak performance when the Rician noise level is large. Moreover, our method demands much less computing time compared with gradient descent method. Our method is also competitive with Liyuan Chen’s convex model. The average denoising performance is better than the convex model. Moreover, our parameter for the model is robust. We even do not change the parameter of the model for denoising different images while other models do adjust the parameter to get a better performance. In this case, our result still outperforms the other two methods.

From the comparison of residual images, we see that the residual from MAP-GD contains more information than the residual from MAP-DCA method. To give a more detailed performance of our method quantitatively, we summarize denoising performance in Table 2.1.

Figure 2.3: Image corrupted by Rician noise.
Figure 2.4: Row 1: Image Cameraman restored with Rician noise $\sigma = 20$. Row 2: Zoom part of image restored by three methods for comparison.

Figure 2.5: Row 1: PeMRI restored under Rician noise with $\sigma = 20$ by three methods. Row 2: Zoom part of image restored by three methods for comparison.
Table 2.1: PSNR values and CPU-time (in second) for Rician noise removal.

<table>
<thead>
<tr>
<th>Images</th>
<th>Methods</th>
<th>$\sigma = 20$</th>
<th>$\sigma = 30$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PSNR</td>
<td>Time(s)</td>
</tr>
<tr>
<td>Cameraman</td>
<td>LY-PD</td>
<td>27.79</td>
<td>0.37</td>
</tr>
<tr>
<td>Cameraman</td>
<td>MAP-GD</td>
<td>27.51</td>
<td>19.7</td>
</tr>
<tr>
<td>Cameraman</td>
<td>MAP-DCA</td>
<td>28.43</td>
<td>1.4</td>
</tr>
<tr>
<td>Brain</td>
<td>LY-PD</td>
<td>26.15</td>
<td>0.38</td>
</tr>
<tr>
<td>Brain</td>
<td>MAP-GD</td>
<td>26.48</td>
<td>13.7</td>
</tr>
<tr>
<td>Brain</td>
<td>MAP-DCA</td>
<td>28.27</td>
<td>1.29</td>
</tr>
<tr>
<td>PeMRI</td>
<td>LY-PD</td>
<td>27.87</td>
<td>0.47</td>
</tr>
<tr>
<td>PeMRI</td>
<td>MAP-GD</td>
<td>27.64</td>
<td>17.2</td>
</tr>
<tr>
<td>PeMRI</td>
<td>MAP-DCA</td>
<td>28.68</td>
<td>1.41</td>
</tr>
</tbody>
</table>

From the table, we observe that our DCA method has got a PSNR on average 0.5\text{dB} higher than the other two methods. We summarize thus that DCA method for denoising Rician noise outperforms the other compared methods.

2.5.2 Image deblurring performance

Not only the MAP model can deal with denoising problem, but also it can be applied for image deblurring. Here we use two blur operators: Gaussian blur and motion blur. For Gaussian blur, we set a window sized $9 \times 9$ with standard deviation 1. For motion blur, we use a length 5 and angle 30 motion blur kernel. The blur kernel is shown in the previous chapter. Then Rician noise with $\sigma = 15$ are added to the blurred image. We compare our DCA method (MAP-DCA) with original gradient descent method [34](MAP-GD), two convex methods including Getreuer’s method [34] (Getreuer) and Liyuan Chen’s new convex method [25](LY-PD).

The performance is in the following figures. Figure 2.6 shows the degraded images with blur and noise. Figure 2.7, 2.8 and 2.9 show the restored image with different methods.
Figure 2.6: image degraded with blur.

Figure 2.7: PSNR values of different methods when deblurring Gaussian blur with Rician noise $\sigma = 15$ for Brain and their residuals of different methods after restoration.
From the PSNR value of deblurring by different methods, we observe that our MAP-DCA method is competitive with the other three methods. All the methods here can give an excellent performance for image deblurring. For all images that we test, our method as well as the other three methods show a strong color contrast of image. This can be shown from the residual image which contains little information. However, for some images, Getreuer’s method is not as robust as others which contains a much enriched residual image. For the two convex models, there exists a bias correction which can be seen from the color constrast of residual images. Thus, we conclude that our method is robust in deblurring medical and natural images corrupted with Rician noise. To give more details of our deblurring performance, please see Table 2.2.

From the table, we see that on average, our DCA method outperforms the other three methods for all tested images for image denoising and deblurring.
Figure 2.9: PSNR values of different methods when deblurring motion blur with Rician noise $\sigma = 15$ for image 'peMRI' and the residuals of different methods after restoration.
Table 2.2: PSNR values by different methods for Rician noise removal under blurry condition.

<table>
<thead>
<tr>
<th>Image</th>
<th>blurType</th>
<th>LY-PD</th>
<th>MAP-GD</th>
<th>Getreuer</th>
<th>MAP-DCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>brain</td>
<td>Gaussian</td>
<td>27.38</td>
<td>28.72</td>
<td>28.92</td>
<td><strong>28.93</strong></td>
</tr>
<tr>
<td></td>
<td>Motion</td>
<td>26.94</td>
<td>27.88</td>
<td><strong>27.89</strong></td>
<td>27.89</td>
</tr>
<tr>
<td>cameraman</td>
<td>Gaussian</td>
<td><strong>25.60</strong></td>
<td>25.35</td>
<td>25.25</td>
<td>25.34</td>
</tr>
<tr>
<td></td>
<td>Motion</td>
<td><strong>25.13</strong></td>
<td>24.90</td>
<td>24.86</td>
<td>24.82</td>
</tr>
<tr>
<td>MRI liver</td>
<td>Gaussian</td>
<td>28.82</td>
<td>28.42</td>
<td>28.21</td>
<td><strong>28.92</strong></td>
</tr>
<tr>
<td></td>
<td>Motion</td>
<td>28.15</td>
<td>28.08</td>
<td>27.89</td>
<td><strong>28.45</strong></td>
</tr>
<tr>
<td>peMRI</td>
<td>Gaussian</td>
<td><strong>28.20</strong></td>
<td>28.06</td>
<td>26.67</td>
<td>28.03</td>
</tr>
<tr>
<td></td>
<td>Motion</td>
<td><strong>28.25</strong></td>
<td>28.13</td>
<td>26.62</td>
<td>28.03</td>
</tr>
<tr>
<td>average</td>
<td></td>
<td>27.31</td>
<td>27.45</td>
<td>27.04</td>
<td><strong>27.56</strong></td>
</tr>
</tbody>
</table>

2.6 Conclusion

In this chapter, we apply DCA method for Rician noise removal with and without blur by MAP non-convex model. DCA can find a better local optimiser than gradient-descent method. How to apply DCA to specific problem is a good research subject since there are infinite methods to decompose the DC functions. For our example, we separate the convex part and concave part directly by putting convex factors in function $G$ and concave factors in function $H$. Sometimes, we need to add an additional term to keep both $G$ and $H$ convex depending on the specific structure of the problem being solved. Comparing with gradient-descent algorithm, DCA method can give a better performance for both denoising and deblurring. Moreover, the computing time used by DCA is much less than that of gradient-descent method. The larger the Rician noise, the better the solution by DCA comparing with other methods. For small Rician noise, DCA method is also competitive with other methods. For different blur kernels, the performance of numerical experiments show that on average, DCA-method performs best among all methods.
Chapter 3

Cauchy noise removal with DC Algorithm

3.1 Introduction

In this chapter, we will use difference of convex algorithm (DCA) to restore images corrupted with Cauchy noise. In many real imaging system and engineering applications, the noise is not the most common Gaussian noise. Sometimes, it shows an impulsive character for the noise distribution. One of the impulsive kind of noise called Cauchy noise usually appears in the real images when photos are taken underwater, or by sonar or radar. Even wireless communication, air turbulence, biomedical imaging and synthetic aperture radar(SAR) imaging can generate Cauchy noise. Cauchy noise belongs to $\alpha$-stable distribution. The $\alpha$-stable distribution can be defined most conveniently by its characteristic function, which is the Fourier transform of its pdf:

$$\phi(t) = e^{j\mu t - \gamma|t|^\alpha}, \quad (3.1)$$

where $-\infty < \mu < \infty$, $\gamma > 0$, and $0 < \alpha \leq 2$. $\alpha$ describes the thickness of the tails of the distribution. When $\alpha = 2$, the distribution corresponds to Gaussian noise while when $\alpha = 1$, the distribution corresponds to Cauchy noise. The dispersion parameter $\gamma$ is a measure of the deviation around the mean. Mathematically, a Cauchy random variable $v$ follows the distribution like:

$$p(v) = \frac{1}{\pi} \frac{\gamma}{\gamma^2 + (v - \epsilon)^2}, \quad (3.2)$$
where \( \gamma > 0 \) and \( \epsilon \) is the spread that the Cauchy distribution is around \( \epsilon \). Suppose \( f \) is the image degraded by Cauchy noise. \( u \) is the original clean image. Then \( f \) is presented as:

\[
 f = u + v,
\]

(3.3)

where \( v \) is Cauchy noise. From now on, we consider \( \epsilon = 0 \) for our following studies. Many methods are proposed for Cauchy noise removal. In [75], the author uses Cauchy-Gaussian mixtures to estimate the parameters of the noise density under various scenarios for symmetric \( \alpha \)-stable process. In [1], the author proposed a method in complex wavelet domain for denoising Cauchy noise. In [24], the author used recursive Bayesian filters to remove Cauchy noise. In [69], the author proposed a variational method for Cauchy denoising. They derived it from MAP Bayesian Statistics. The model is:

\[
\inf_{u \in BV(\Omega)} J(u) + \frac{\lambda}{2} \left( \int \log(\gamma^2 + (u - f)^2) + \mu \|u - u^0\|_2^2 \right). \tag{3.4}
\]

They add a term \( \mu \|u - u^0\|_2^2 \) to make the model convex. However, the term \( u^0 \) is a median filter of \( u \) which is hard to understand. In our paper, we try to solve the MAP (Maximum A Posteriori) non-convex model directly by DCA (Difference of Convex Algorithm) theory. We will solve directly the non-convex model derived from Bayesian theory:

\[
\inf_{u \in BV(\Omega)} J(u) + \frac{\lambda}{2} \int \log(\gamma^2 + (u - f)^2). \tag{3.5}
\]

3.2 DCA method for solving MAP variational model for Cauchy noise

In this section, we deduct the MAP model by Bayesian Statistics. Then we use DCA method to solve the non-convex model. We prove the existence of the solution of the MAP model and the convergence of the numerical solution.
### 3.2.1 MAP model for Cauchy noise removal

As a zero-centered Cauchy noise $v$, the density function is:

$$g_V(v) = \frac{1}{\pi \gamma^2 + v^2}.$$  \hfill (3.6)

We want to recover the original clean image $u$ given the noisy image $f$. Mathematically, we maximize the probability of $P(u|f)$. Similar with chapter 1, we have:

$$\max \log P(u|f) = \min -\log(P(f|u)) - \log(P(u)),$$

$$= \min E(u),$$

where

$$E(u) := \int_{\Omega} |Du| + \frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (u - f)^2) dx,$$  \hfill (3.7)

and $\int_{\Omega} |Du|$ is defined as total variation in BV space. In [69], the author has already proved the existence of the minimizer for model (3.7). However, since the model is non-convex, the author who proposed the variational model (3.7) has not used this model directly. She proposed a convex model by adding another data fidelity term. Now our contribution in this chapter is to solve directly the non-convex model by DCA method.

### 3.2.2 DCA solution

DCA method has been introduced in our previous chapter. Here we will use the definition and the properties directly for this chapter. DCA method is based on solving a problem which is composed by the difference of two convex functions. Then we give the DC function of problem (3.7):

$$\inf_{u \in BV(\Omega)} E(u) := G(u) - H(u),$$  \hfill (3.8)

where

$$G(u) = \int_{\Omega} |Du| + c\|u\|^2,$$  \hfill (3.9)

$$H(u) = -\frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (u - f)^2) dx + c\|u\|^2.$$  \hfill (3.10)
We add in the same time the item $c\|u\|^2$ for $G(u)$ and $H(u)$ in order to ensure the strict convexity of the two functions. We can prove the convexity of $G(u), H(u)$ by some mild condition for parameter $c$.

**Lemma 5.** $G(u)$ is convex for $u \in BV(\Omega)$, $H(u)$ is convex if $c \geq \frac{\lambda}{2\gamma^2}$.

**Proof.** $G(u)$ is obviously convex since the sum of the two norm is convex by the triangle inequality. Now we begin to prove the convexity of $H(u)$. We compute the first and second order derivatives of $H(u)$:

$$H'(u) = -\frac{\lambda(u-f)}{\gamma^2 + (u-f)^2} + 2cu.$$  
$$H''(u) = \frac{\lambda((u-f)^2 - \gamma^2)}{(\gamma^2 + (u-f)^2)^2} + 2c.$$  

In order to let $H(u)$ convex, we must make $H(u) \geq 0$. Then by calculation, we have:

$$c \geq \frac{1}{2} \frac{\lambda(\gamma^2 - (u-f)^2)}{(\gamma^2 + (u-f)^2)^2}.$$  

The maximum value of item $\frac{1}{2} \frac{\lambda(\gamma^2 - (u-f)^2)}{(\gamma^2 + (u-f)^2)^2}$ is when $u - f = 0$. Then we have:

$$c \geq \frac{1}{2} \frac{\lambda \gamma^2}{\gamma^4} = \frac{\lambda}{2\gamma^2}.$$  

We finish the proof.

### 3.2.3 Convergence analysis and numerical solution for DCA method

Now we introduce the DCA approach for solving $u$ in (3.7). $H(u)$ can be replaces by its affine minimization around $u^k$:

$$H(u) = H(u^k) + \langle u - u^k, v^k \rangle, \quad v^k \in \partial H(u^k).$$  

Then we get $v^k$ by making first order derivative to $H(u)$:

$$v^k = -\frac{\lambda(u^k-f)}{\gamma^2 + (u^k-f)^2} + 2cu^k.$$  

Then since \( u^k = \arg \min G(u) - \langle v^k, u \rangle \), we have:

\[
u^k = \arg \min TV(u) + c\|u\|^2 - \langle v^k, u \rangle.
\] (3.13)

By dual transform of TV norm, \( u^k \) in (3.13) is equivalent with:

\[
u^k = \arg \min \max_{u, \|p\| \leq 1} -\langle u, \text{div} \ p \rangle + c\|u\|^2 - \langle v^k, u \rangle.
\] (3.14)

Then the original problem can be transferred to a saddle point problem:

\[
\min_{u, u^k} \max_{u, \|p\| \leq 1} E(u, p),
\] (3.15)

where

\[
E(u, p) = -\langle u, \text{div} \ p \rangle + c\|u\|^2 - \langle v^k, u \rangle.
\]

The existence of solution of the saddle point problem (3.15) can be proved easily by satisfying the four conditions. We can refer to Lemma 2 in Chapter 2. The uniqueness of the solution is proved in [69] under mild condition.

**Lemma 6.** Let \( u^k \) be the sequence generated by the DCA algorithm for solving (3.7) by decomposition (3.9). Then we have:

\[
F(u^k) - F(u^{k+1}) \geq 0.
\] (3.16)

We can refer to the proof in Lemma 3 in chapter 2.

**Theorem 4.** Any limit point \( u^* \) of \( u^n \) satisfies a weak first-order optimality condition:

\[
0 \in \partial |Du^*| + \frac{\lambda(u^* - f)}{\gamma^2 + (u^* - f)^2}.
\] (3.17)

**Proof.** The sequence \( u^k \) is bounded since it represents the image pixel, then based on Bolzano-Weierstrass theorem that it exists a subsequence of \( u^k \) denoted as \( u^{n_k} \), converging to \( u^* \). The optimality condition at the \( n_k \) step of the DCA is:

\[
0 \in \partial |Du^{n_k}| + \frac{\lambda(u^{n_k} - f)}{\gamma^2 + (u^{n_k} - f)^2}.
\] (3.18)

As \( u^{n_k} \rightarrow u^* \), we have \( Du^{n_k} \rightarrow Du^* \), since \( |D| < \infty \). Since \( \gamma > 0 \) and \( u^{n_k} \) is bounded, \( \frac{\lambda(u^{n_k} - f)}{\gamma^2 + (u^{n_k} - f)^2} \) is not infinite, such that \(-\infty < \frac{\lambda(u^{n_k} - f)}{\gamma^2 + (u^{n_k} - f)^2} < \infty \). Then we have \( \frac{\lambda(u^{n_k} - f)}{\gamma^2 + (u^{n_k} - f)^2} \rightarrow \frac{\lambda(u^* - f)}{\gamma^2 + (u^* - f)^2} \). Adding the two items together, we proved the theorem. 

35
Thus, the method can be solved numerically,

\[ p^{k+1} = \text{arg max}_{p, \|p\| \leq 1} \gamma \langle \nabla u^k, p \rangle - \frac{1}{2\beta} \|p - p^m\|_2^2; \quad (3.19) \]

\[ u^{k+1} = \text{arg min}_{u \in \Omega} - \langle u, \text{div} p^{k+1} \rangle + c \|u\|^2 - \langle v_k, u \rangle + \frac{1}{2\tau} \|u - u^k\|_2^2. \quad (3.20) \]

The solutions of (3.19), (3.20) are:

\[ \tilde{p}^{k+1} = p^k + \beta \gamma \nabla u^k, \quad p^{k+1} = \frac{\tilde{p}^{k+1}}{\max(1, |\tilde{p}^{k+1}|)}; \quad (3.21) \]

\[ u^{k+1} = \frac{u^k + \tau (u_k + \text{div} p)}{2c\tau + 1}. \quad (3.22) \]

To summarize, we start with an initial value of image \( u \), the proposed approach iterates as in Algorithm 3. The stopping criteria for outer and inner iterations are:

\[ \frac{\|F u^k - F u^{k+1}\|_2}{\|F u^k\|_2} < \text{tol}_{\text{out}}; \quad (3.23) \]

\[ \frac{\|u^k - u^{k+1}\|_2}{\|u^k\|_2} < \text{tol}_{\text{in}}. \quad (3.24) \]

where we set \( \text{tol}_{\text{in}}, \text{tol}_{\text{out}} = 10^{-4} \).

**Algorithm 3** Algorithm for solving MAP model for Cauchy noise removal

1: Fixed \( \gamma, \tau, \beta, \text{maxOutIterNum}, \text{maxInIterNum}, \text{tol}_{\text{in}}, \text{tol}_{\text{out}} \). Initiate: \( u_0 = f, p = (0, 0, \cdots, 0) \).

2: for \( k = 1 \) to \( \text{maxOutIterNum} \) do

3: \( v^k = \frac{\lambda (u^k - f)}{1 + (u^k - f)^2} + 2cu^k. \)

4: for \( m = 1 \) to \( \text{maxInIterNum} \) do

5: calculate \( p^{k+1} \) by (3.21).

6: calculate \( u^{k+1} \) by (3.22).

7: If \( \frac{\|u^k - u^{k+1}\|_2}{\|u_k\|_2} < \text{tol}_{\text{in}}, u^k = u^{k+1}, \) break;

8: end for

9: If \( \frac{\|F u^k - F u^{k+1}\|_2}{\|F u_k\|_2} < \text{tol}_{\text{out}}, \) break;

10: end for
3.3 Cauchy noise removal under blurry condition

In this section, we develop MAP model for Cauchy noise removal with blurry condition. The MAP model for denoising and deblurring in the same time is:

$$\inf_{u \in BV(\Omega)} \int_{\Omega} |Du| + \frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (Ku - f)^2) dx.$$  \hfill (3.25)

3.3.1 Existence of solution for Cauchy MAP model under blurry condition

In the following theorem, we prove the existence of the solution for model (3.25).

Theorem 5. Let $f$ be in $L^\infty(\Omega)$, let $K \in L^2(\Omega)$, $K$ is non-negative, non-empty and linear since it is a blurring operator. Then the solution of model (3.25) exists.

Proof. First, we can prove the minimization problem (3.25) is bounded from below. Since a minimizing sequence $u_n$ is in the set $BV(\Omega)$ for (3.25), assume $a \leq u_n \leq b$ since $u_n$ is the image pixel. Then from the expression of $F(u_n)$, we have: $F(u_n) \geq 0$ and since we proved already $F(u^k) \geq F(u^{k+1})$, then we have $0 \leq F(u_n) \leq F(u_0)$. Then $F(u_n)$ is bounded. Since

$$F(u_n) = |Du_n| + \frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (Ku_n - f)^2) dx$$

and $\frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (Ku_n - f)^2) dx$ arrives at its minimum at $Ku = f$, then we have $|Du_n|$ is bounded. Therefore, we have that $u_n$ is bounded in $BV(\Omega)$ and by the compactness of $L^1(\Omega)$ in BV, there exists a subsequence $\{u_{n_k}\}$ which converges to $u$ in $L^1(\Omega)$. By the lower semi-continuity of the BV norm, $TV(u) \leq \liminf TV(u_{n_k})$ in $L^1(\Omega)$, where $TV(u) = |Du|$. Since the operator $K$ is continuous, $Ku_{n_k}$ also converges weakly to $Ku$ in $L^1(\Omega)$. Based on Fatou’s lemma, $u$ minimizes (3.25). Then we finish the proof. \hfill \Box

Evidently, the proof also works for denoising case when $K = Id$.  

37
3.3.2 Numerical algorithm for Cauchy noise removal under blurry condition

Now we give the DC functions for denoising and deblurring image with Cauchy noise. The DC functions for the corresponded MAP model is:

\[
G(u) = \int_{\Omega} |Du| + c\|u\|^2, \quad (3.26)
\]

\[
H(u) = -\frac{\lambda}{2} \int_{\Omega} \log(\gamma^2 + (Ku - f)^2) dx + c\|u\|^2. \quad (3.27)
\]

The numerical algorithm is similar with denoising case. We give directly the solutions of \(u^k, p^k\) directly.

\[
\tilde{p}^{k+1} = p^k + \beta \gamma \nabla u^k, \quad p^{k+1} = \frac{\tilde{p}^{k+1}}{\max(1, |\tilde{p}^{k+1}|)}; \quad (3.28)
\]

\[
u^{k+1} = u^k + \tau (v^k + \text{div} p) \frac{2c\tau + 1}{2c\tau + 1}, \quad (3.29)
\]

where \(v^k = -\frac{\lambda K^T(Ku - f)}{\gamma^2 + (Ku - f)^2} + 2cu^k\). The condition for the convergence of the numerical primal-dual algorithm is \(\beta \tau \|\nabla\|^2 \leq 1\). Please refer to [20] for a proof.

3.4 Numerical results of image denoising and deblurring

Now we give the performance of MAP-DCA method for Cauchy noise removal. We first focus on the denoising case. Then we make experiments for image denoising and deblurring simultaneously. We compare our method with convex MAP model (MAP-Convex) which is illustrated in [69]. We also compare with classical ROF model (ROF). We use the peak signal noise ratio (PSNR) value to measure image quality of restoration. The PSNR formula has been defined in Chapter 2. The stopping criteria is listed in Algorithm 3 for both denoising and deblurring experiments. All simulations are run in MATLAB R2013a.
Table 3.1: PSNR value of Cauchy noise removal.

<table>
<thead>
<tr>
<th>Images</th>
<th>Noisy</th>
<th>ROF</th>
<th>MAP-convex</th>
<th>MAP-DCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>19.03</td>
<td>24.61</td>
<td>28.08</td>
<td>28.41</td>
</tr>
<tr>
<td>Parrot</td>
<td>19.13</td>
<td>24.56</td>
<td>28.96</td>
<td>28.79</td>
</tr>
<tr>
<td>Peppers</td>
<td>19.12</td>
<td>25.73</td>
<td>30.51</td>
<td>30.33</td>
</tr>
<tr>
<td>Lena</td>
<td>19.17</td>
<td>26.11</td>
<td>31.06</td>
<td>30.53</td>
</tr>
<tr>
<td>Average</td>
<td>19.11</td>
<td>25.25</td>
<td>29.65</td>
<td>29.52</td>
</tr>
</tbody>
</table>

3.4.1 Image denoising performance

Now we give the performance of image denoising. The standard Cauchy random variable can be generated by the division of two standard normal variables, such that:

\[ f = u + v = u + \xi \frac{\eta_1}{\eta_2}, \]

where \( v \) is the variable which follows Cauchy distribution. The Cauchy distribution can be decomposed by the division of two variables with standard Gaussian distribution. Here \( \xi \) represents the noise level and \( \eta_1, \eta_2 \) follow the Gaussian distribution with mean 0 and variance 1. The performance of image denoising with our method and with comparing methods: MAP-convex and ROF are in Figure 3.1, 3.2 and Table 3.1, 3.2 under different noise level.

From the performance, we can see that our method and the MAP-convex method both outperforms classical ROF model. Our method is very competitive with the convex method. The average of the PSNR for four images are quite close to convex method. Then the advantage of our model is that our model is more simple. We do not need to calculate the median filter. Thus, our model is more straightforward and explicit to apply.
Figure 3.1: Cauchy noise removal for $\xi = 0.02$. 

<table>
<thead>
<tr>
<th>Noisy</th>
<th>ROF</th>
<th>MAP-convex</th>
<th>MAP-DCA</th>
</tr>
</thead>
</table>


Figure 3.2: Cauchy noise removal for $\xi = 0.04$. 
Table 3.2: PSNR value of Cauchy noise removal.

<table>
<thead>
<tr>
<th>Images</th>
<th>Noisy</th>
<th>ROF</th>
<th>MAP-convex</th>
<th>MAP-DCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>16.21</td>
<td>23.53</td>
<td>26.47</td>
<td>26.76</td>
</tr>
<tr>
<td>Parrot</td>
<td>16.21</td>
<td>23.10</td>
<td>27.00</td>
<td>26.82</td>
</tr>
<tr>
<td>Peppers</td>
<td>16.21</td>
<td>24.42</td>
<td>28.46</td>
<td>27.87</td>
</tr>
<tr>
<td>Lena</td>
<td>16.16</td>
<td>25.01</td>
<td>28.82</td>
<td>28.41</td>
</tr>
<tr>
<td>Average</td>
<td>16.20</td>
<td>24.01</td>
<td>27.69</td>
<td>27.47</td>
</tr>
</tbody>
</table>

Table 3.3: PSNR value of Cauchy noise removal under Gaussian blur.

<table>
<thead>
<tr>
<th>Images</th>
<th>Noisy</th>
<th>ROF</th>
<th>MAP-convex</th>
<th>MAP-DCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>18.32</td>
<td>24.23</td>
<td>26.13</td>
<td>26.16</td>
</tr>
<tr>
<td>Parrot</td>
<td>18.33</td>
<td>24.31</td>
<td>26.70</td>
<td>26.80</td>
</tr>
<tr>
<td>Peppers</td>
<td>19.12</td>
<td>25.73</td>
<td>27.25</td>
<td>27.13</td>
</tr>
<tr>
<td>Lena</td>
<td>18.76</td>
<td>26.60</td>
<td>29.00</td>
<td>28.97</td>
</tr>
<tr>
<td>Average</td>
<td>18.63</td>
<td>25.22</td>
<td>27.27</td>
<td>27.27</td>
</tr>
</tbody>
</table>

3.4.2 Image deblurring and denoising performance

Now we show performance of restoring images corrupted by Cauchy noise and Gaussian blur. The Gaussian blur is set with a window sized $9 \times 9$ and standard deviation of 1. We add Gaussian noise with level $\xi = 0.02$ after the blurring process. We still compare our MAP-DCA method with MAP-convex method, ROF method. In Figure 3.3 and Table 3.3, we give the performance of image deblurring and denoising.

From the PSNR value in the table, we show that our DCA method can generate a robust performance of Cauchy noise removal under blurry condition. Our method and the convex MAP method are competitive and better than the ROF method.
Figure 3.3: Cauchy noise removal under Gaussian blur for variance 1 with $\xi = 0.02$. 
3.5 Conclusion

In this chapter, we apply DCA method to solve a non convex variational model for Cauchy noise removal. We prove the existence of the solution for the non-convex model and the convergence of the numerical solution. With DCA method, Cauchy noise can be well reduced even with blurry condition. Compared with other methods, our method can give a competitive performance. Comparing with the convex model, our model is more simple and easy to implement.
Chapter 4

A new image quality metric
Sharpness Q-Index and its application

4.1 Introduction and background

Besides image restoration which we study on in our previous two chapters, image quality assessment is another very important research subject in image processing. The applications of image quality assessment can include: supervising the image quality in real-time for video and website, assessing image restoration performance and optimizing parameters for image restoration models. Intuitively, we can judge the image quality after restoration by eyes which is called subjective quality assessment. However, this is inaccurate and time-consuming. Instead, research is assessed by objective quality assessment. The objective quality assessment is based on a special metric which uses the information of restored image, corrupted image before restoration and possibly the original clean image to give a value. The bigger the value of the metric, the better the image quality. There are many metrics for image quality assessment. The methods to assess restored image quality from a corrupted image can be classified depending on whether we know the information of the original clean image. Generally we divide the assessment methods into three categories: full reference image quality assessment, reduced reference image quality assessment and no reference image quality assessment.
Full reference is a reference which uses the information of original image. Typical examples are the famous metric: the peak-signal-to-noise ratio (PSNR) [43, 44], the structural similarity index measure (SSIM) [85] as well as the mean square error (MSE). Interesting readers can refer to [47, 70] for a complete discussion of the full reference metric. For reduced reference, only partial information of the original image or only a set of local features are used to calculate the metric [86]. For no reference quality assessment, there is no original image information for reference. No reference assessment is only an absolute value which is calculated based on some features of image. A detailed overview of existing no reference metrics is given in [32]. Sharpness metric is one of no-reference metrics which is defined by the different tones or colors of the boundaries. There are ways to measure sharpness. Sharpness metrics can be measured by gradient function. In [31], Marziliano and Ong et al. measure the image based on smoothing effects of edge blur. In [81], a locally-adaptive iterative edge refinement algorithm was proposed which can be regarded as sharpness metric and can give accuracy to edge detection especially under highly blurred image. In [94], the author proposed a sharpness metric based on SVD (singular value decomposition). The sharpness metric will drop either when there is blur or noise which corrupts the image. In [32], measured just-noticeable blurs (JNBs) is proposed for measuring sharpness. In [59] Narvekar and Karam estimated the sharpness of an image as the cumulative probability of detecting blur at an edge (CPBD). In our thesis, we propose a new no-reference image quality metric called Sharpness q-Index (SQ-Index) which is related to previous work of G, Blanchet and L, Moisan [50] who have proposed three similar metrics: Global Phase Coherence, Sharpness Index and S-Index. The new metric can serve for parameter selection for various image restoration models. In the following sections, we introduce some related work with respect to SQ-index. Then we introduce the definition of SQ-index and its mathematical properties. In order to illustrate how to use SQ-Index to choose parameters for variational model, we introduce a constrained model and propose a new algorithm to solve it. Then we
show the performance of Gaussian noise estimation, image denoising and deblurring by constrained model with parameters selected by SQ-index and compare our method with other parameter selection methods.

4.2 Related work

In Fourier domain, an image includes two components: the amplitude and the phase. The discrete Fourier transform of $u$ is:

$$\hat{u}(\epsilon) = \sum_{x \in \Omega} u(x) e^{-\frac{2\pi i}{n} \langle x, \epsilon \rangle}$$

(4.1)

$$= |\hat{u}(\epsilon)| \cdot e^{i\varphi(\epsilon)}.$$  

(4.2)

where $|\hat{u}(\epsilon)|$ is the amplitude and $\varphi(\epsilon)$ is the phase. To understand the influence of $\varphi$, we can add phase information to the original image as:

$$u_{\varphi}(x) = \frac{1}{n^2} \sum_{\epsilon \in R} |\hat{u}(\epsilon)| \cdot e^{i\psi(\epsilon)},$$

(4.3)

where $\psi = \varphi + \epsilon S$, $\epsilon$ is a constant between $[0, 1]$ and $S$ is a random variable uniformly distributed on $(-\pi, \pi)$. Based on the theory of Oppenheim and Lim [62], it was discovered that the phase of the Fourier transform of an image is important in keeping geometry of an image, especially its contours. Figure 4.2 shows how phase information affects images.

Morrone and Owens [58] proposed the definition of Maximum Phase Congruency which is efficient for detecting edges of images and Kovesi [46] improved the definition. Later, Wang and Simoncelli [84] defined local phase coherence based on wavelet phase transform which can detect blur. All the above papers show that the phase component plays an important role in detecting image quality.

4.2.1 Preliminary

Before going on the theory of global phase coherence, let us first define some notations. Suppose that the gray-level images $u$ is defined on a discrete $M \times N$ rectangular
domain $\Omega$, where $\Omega$ is defined as:

$$
\Omega = \mathbb{Z}^2 \cap \left( [-\frac{M}{2}, \frac{M}{2}] \times [-\frac{N}{2}, \frac{N}{2}] \right).
$$

The $\Omega$-periodicity of $u$ is the image $\dot{u} : \mathbb{Z}^2 \to \mathbb{R}$ that extends $u$ to the whole surface $\mathbb{Z}^2$ by:

$$
\forall (i, j) \in \mathbb{Z}^2, \forall (x, y) \in \Omega, \dot{u}(x + iM, y + jN) = u(x, y). \quad (4.4)
$$

The gradient of $\dot{u}$ is defined by: $\forall (x, y) \in \mathbb{Z}^2$,

$$
\nabla \dot{u}(x, y) = \begin{pmatrix}
\frac{\partial}{\partial x} \dot{u}(x, y) \\
\frac{\partial}{\partial y} \dot{u}(x, y)
\end{pmatrix} = \begin{pmatrix}
\dot{u}(x + 1, y) - \dot{u}(x, y) \\
\dot{u}(x, y + 1) - \dot{u}(x, y)
\end{pmatrix}. \quad (4.5)
$$

The autocorrelation of $\nabla \dot{u}$ is defined by: $\Gamma : \Omega \to \mathbb{R}^{2 \times 2}$

$$
\Gamma(z) = \begin{pmatrix}
\Gamma_{xx}(z) & \Gamma_{xy}(z) \\
\Gamma_{yx}(z) & \Gamma_{yy}(z)
\end{pmatrix} = \sum_{y \in \Omega} \nabla \dot{u}(y) \nabla \dot{u}(y + z)^T. \quad (4.6)
$$

### 4.2.2 Global Phase Coherence

The global phase coherence (GPC) is defined by measuring how much the geometry of an image is changed when the phase information is changed or when the phase information is made random. To measure the phase information component $\varphi$, a
likelihood measurement is proposed based on total variation. They define GPC as a probability density function \( p(u) \). For different phase functions \( \psi \), some of them will give a more possible likelihood \( p(u_\psi) > p(u) \), but most of them will give a less possible likelihood \( p(u_\psi) < p(u) \). The definition of GPC is:

\[
\text{GPC}(u) = -\log_{10} \mathbb{P}(\text{TV}(u_\psi) \leq \text{TV}(u))
\]

(4.7)

where \( \mathbb{P} \) is the probability. TV is the TV norm defined by:

\[
\text{TV}(u) = \sum_{x \in \Omega} |\partial_x \hat{u}(x)| + |\partial_y \hat{u}(x)|.
\]

(4.8)

Here \( \hat{u} \) is \( \Omega \)-periodicity of \( u \) defined by (4.4) and \( u_\psi \) is the random phase image: the original image with a random phase noise \([50]\). Intuitively, the TV-norm of a random phase image will be much larger than the TV-norm of its original image. By experiment in \([12]\), GPC tends to decrease when image is corrupted with blur or noise. GPC can be used not only for image quality assessment by combining it with other measures \([82]\), it can also be used for blind-deblurring. Paper \([12]\) gives an example.

### 4.2.3 Sharpness Index

A new measure called Sharpness Index derived of Global Phase Coherence is introduced in \([11]\) when \( u_\psi \) is replaced by \( u \ast W \): original image \( u \) convoluted with a Gaussian white noise \( W \). Then Sharpness Index was proposed as follows:

\[
\text{SI}(u) = -\log_{10}(\mathbb{P}(\text{TV}(u \ast W) \leq \text{TV}(u)))
\]

(4.9)

where \( W \) is a Gaussian noise with:

\[
\forall x \in \Omega, W(x) \sim N(0, |\Omega|^{-1}).
\]

The difference of Sharpness Index with Global Phase Coherence is that the random phase image is replaced by an image convoluted with a Gaussian noise \( W \), then the
calculation of expectation and variance of $TV(u * W)$ can be explicit. Blanchet G, Moisan L. made an assumption and then proved that r.v. $TV(u * W)$ is approximately Gaussian [50]. Then the Sharpness Index is re-defined as:

$$SI(u) = -\log_{10} \Phi\left(\frac{\mu - TV(u)}{\sigma}\right)$$

(4.10)

where $\Phi$ is the standard Gaussian probability density function: $\Phi(t) = \frac{1}{\sqrt{2\pi}} \int_{t}^{\infty} e^{-\frac{s^2}{2}} ds$.

They also give explicit formulas for the expectation $\mu$ and variance $\sigma^2$ of $TV(u * W)$:

$$\mu = (\alpha_x + \alpha_y)\sqrt{\frac{2}{\pi}} \sqrt{MN},$$

(4.11)

$$\sigma^2 = \frac{2}{\pi} \sum_{z \in \Omega} \alpha_x^2 w(\frac{\Gamma_{xx}(z)}{\alpha_x^2}) + 2\alpha_x \alpha_y w(\frac{\Gamma_{xy}(z)}{\alpha_x \alpha_y}) + \alpha_y^2 w(\frac{\Gamma_{yy}(z)}{\alpha_y^2}),$$

(4.12)

where $\alpha_x = ||\partial_x \hat{u}||_2$ and $\alpha_y = ||\partial_y \hat{u}||_2$, $w$ is the function:

$$\forall t \in [-1, 1], \quad w(t) = t \cdot \arcsin(t) + \sqrt{1 - t^2} - 1.$$  

(4.13)

The Sharpness Index is very close to Global Phase Coherence. They only differ by a multiplicative Rayleigh Noise. The proof is given in [50]. Therefore, Sharpness Index is more practical than Global Phase Coherence since it has an explicit formula instead of heavy Monte-Carlo simulation.

### 4.2.4 S-Index

There is a simplified version of Sharpness Index. The $w(t)$ in (4.13) in Sharpness Index can be approximated by $t^2$. The proof was given in [50]. Therefore, the S-Index is defined almost the same with Sharpness Index except:

$$\sigma^2_a = \frac{1}{\pi} \left( \frac{||\Gamma_{xx}(z)||^2}{\alpha_x^2} + 2 \frac{||\Gamma_{xy}(z)||^2}{\alpha_x \alpha_y} + \frac{||\Gamma_{yy}(z)||^2}{\alpha_y^2} \right).$$

(4.14)

Then the S-Index makes the calculation much faster.

### 4.3 Sharpness Q-Index

In this section, we propose a new measure of global phase coherence: Sharpness $q$-Index (SQ-Index) which is coherent with Sharpness Index when $q = 1$. It is defined
as:

$$SI_q(u) = -\log_{10} \Phi\left( \frac{\mu - TV_q(u)}{\sigma} \right)$$  \hspace{1cm} (4.15)$$

$TV_q(u)$ is defined as:

$$TV_q(u) = \sum_{x \in \Omega} |\partial_x \hat{u}(x)|^q + |\partial_y \hat{u}(x)|^q.$$  \hspace{1cm} (4.16)

with $\mu = (E(TV_q(u * W))$ and $\sigma = \text{Var}(TV_q(u * W))$.

where $\Phi$ and $W$ are defined the same way as in Sharpness Index. In order to use SQ-Index explicitly, we also need to calculate the expectation and variance of $TV_q(u * W)$.

4.3.1 Calculation of $E(TV_q(u * W))$

The calculation of $E(TV_q(u * W))$ is given in the lemma below. We can see, it is consistent with Sharpness Index when $q = 1$.

**Lemma 7.** Assume that $u : \Omega \to \mathbb{R}$ is the clean image, $W$ is the Gaussian white noise that $W \sim N(0, |\Omega|^{-\frac{1}{2}})$, then we have

$$E(TV_q(u * W)) = \frac{\Gamma\left(\frac{q+1}{2}\right)}{\sqrt{\pi}} \left( \frac{2}{|\Omega|} \right)^{\frac{q}{2}} (\alpha_x^q + \alpha_y^q),$$  \hspace{1cm} (4.17)

where

$$\alpha_x = \|\partial_x \hat{u}\|_2 = \sqrt{\sum_{x,y \in \Omega} |\hat{u}(x+1,y) - \hat{u}(x,y)|^2}$$

and

$$\alpha_y = \|\partial_y \hat{u}\|_2 = \sqrt{\sum_{x,y \in \Omega} |\hat{u}(x,y+1) - \hat{u}(x,y)|^2}.$$  

**Proof.** Note that if $X$ follows Gaussian distribution with standard variation $\sigma$, then

$$E(|X|^q) = \sigma^q \cdot \frac{2^\frac{q}{2} \Gamma\left(\frac{q+1}{2}\right)}{\sqrt{\pi}}.$$  

Since $U = u * W$, we have $\partial_x \hat{U} = (\partial_x \hat{u}) * W$ by linearity. Thus, $\partial_x \hat{U}$ is a stationary
Gaussian field with a marginal distribution of zero mean and its variance is:

\[
\mathbb{E}(|\partial_x \dot{U}(x)|^2) = \frac{1}{MN} \sum_{y \in \Omega} (\partial_x \dot{u}(x-y))^2 \\
= \frac{1}{MN} \alpha_x^2 \\
= \frac{\alpha_x^2}{|\Omega|}.
\]

As a Gaussian approximation of \(\partial_x \dot{U}(x)\), we have:

\[
\mathbb{E}(|\partial_x \dot{U}(x)|^q) = \frac{2^q \Gamma\left(\frac{q+1}{2}\right)}{\sqrt{\pi}} \mathbb{E}(|\partial_x \dot{U}(x)|^2) \frac{1}{\alpha_x^q}.
\]

### 4.3.2 Calculation of \(\text{Var}(TV_q(u \ast W))\)

Now we need to calculate \(\sigma_a^2 = \mathbb{E}(TV_q(U)^2) - (\mathbb{E}(TV_q(U)))^2\). First we calculate \(\mathbb{E}(TV_q(U)^2)\). From the definition of TV_q, \(\mathbb{E}(TV_q(U)^2)\) can be expressed as:

\[
\mathbb{E}(TV_q(U)^2) = \mathbb{E}\left(\sum_{x \in \Omega} |\partial_x \dot{U}(x)|^q + |\partial_y \dot{U}(x)|^q \right)^2
\]

\[
= \sum_{x,y \in \Omega} \mathbb{E}(|\partial_x \dot{U}(x)\partial_x \dot{U}(y)|^q) + \mathbb{E}(|\partial_x \dot{U}(x)\partial_y \dot{U}(y)|^q)
\]

\[
+ \mathbb{E}(|\partial_y \dot{U}(x)\partial_x \dot{U}(y)|^q) + \mathbb{E}(|\partial_y \dot{U}(x)\partial_y \dot{U}(y)|^q).
\]

(4.18)

We set \(z = y - x\) and use the stationary of \(\nabla \dot{U}\), then we have:

\[
\mathbb{E}(TV_q(U)^2) = MN \sum_{z \in \Omega} \mathbb{E}\{(|\partial_x \dot{U}(0)\partial_x \dot{U}(z)|^q) + \mathbb{E}(|\partial_x \dot{U}(0)\partial_y \dot{U}(z)|^q)
\]

\[
+ \mathbb{E}(|\partial_y \dot{U}(0)\partial_x \dot{U}(z)|^q) + \mathbb{E}(|\partial_y \dot{U}(0)\partial_y \dot{U}(z)|^q)\}.
\]

(4.19)

Each term of this sum has the form \(\mathbb{E}(|XY|^q)\) where \((X,Y)\) is a zero mean 2-dimensional Gaussian vector with covariance matrix:

\[
\begin{pmatrix}
\mathbb{E}(X^2) & \mathbb{E}(XY) \\
\mathbb{E}(XY) & \mathbb{E}(Y^2)
\end{pmatrix}.
\]
For instance, we have $X = \partial_x \dot{U}(0)$ and $Y = \partial_y \dot{U}(z)$, thus

$$E(XY) = E \left( \sum_{x,y} \partial_x \dot{u}(x) \partial_y \dot{u}(y) W(x) W(y) \right)$$

$$= \frac{1}{MN} \sum_{x \in \Omega} \partial_x \dot{u}(x) \partial_y \dot{u}(z + x)$$

$$= \frac{1}{MN} \Gamma_{xy}(z)$$

and the covariance matrix $(X, Y)$ is

$$\frac{1}{MN} \begin{pmatrix} \alpha_x^2 & \Gamma_{xy}(z) \\ \Gamma_{xy}(z) & \alpha_y^2 \end{pmatrix}$$

where $\Gamma_{xy}(z) = \sum_{x \in \Omega} \partial_x \dot{u}(x) \partial_y \dot{u}(z + x)$.

The calculation of $E(|XY|^q)$ is in the lemma below.

**Lemma 8.** Define $X = \partial_x \dot{U}(0)$ and $Y = \partial_y \dot{U}(z)$, $E(X^2) = a^2$, $E(Y^2) = b^2$, $E(XY) = ab \sin \theta$. Then we have:

$$E(|XY|^q) \simeq 2^q (\cos \theta)^{2q+1} \Gamma(q + 1) |ab|^q \pi \left(2A_q(0) + A''_q(0) \theta^2\right)$$

where

$$A_q(\theta) := \int_0^{+\infty} \frac{t^q}{(1 + t^2 - 2t \sin \theta)^{q+1}} dt.$$

Before calculating $E(|XY|^q)$, let us first prove a lemma:

**Lemma 9.** Assume that $Z = (X, Y)^T$ is a Gaussian random vector with zero mean and covariance matrix

$$E(ZZ^T) = \begin{pmatrix} 1 & \sin \theta \\ \sin \theta & 1 \end{pmatrix},$$

where $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. Then we have:

$$E(|XY|^q) = \frac{2^{q+1}(\cos \theta)^{2q+1} \Gamma(q + 1)}{\pi} A_q(\theta),$$

53
where
\[ A_q(\theta) := \int_0^{+\infty} \frac{t^q}{(1 + t^2 - 2t \sin \theta)^{q+1}} dt, \]
and
\[ \mathbb{E}(|XY|^q) \simeq \frac{2^{q+1}(\cos \theta)^{2q+1}\Gamma(q+1)}{\pi} (A_q(0) + \frac{A''_q(0)}{2} \theta^2). \]

**Proof.** Let \( \theta \in (-\frac{\pi}{2}, \frac{\pi}{2}) \) and \( C = \begin{pmatrix} 1 & \sin \theta \\ \sin \theta & 1 \end{pmatrix} \). Then we have:
\[ C^{-1} = \frac{1}{\cos^2 \theta} \begin{pmatrix} 1 & -\sin \theta \\ -\sin \theta & 1 \end{pmatrix}. \]

Thus,
\[ \mathbb{E}(|XY|^q) = \frac{1}{2\pi \cos \theta} \int_{\mathbb{R}^2} |xy|^q \exp \left( -\frac{x^2 + y^2 - 2xy \sin \theta}{2\cos^2 \theta} \right) dxdy. \]

By symmetry, this can be rewritten as
\[ \mathbb{E}(|XY|^q) = \frac{I_q(\theta) + I_q(-\theta)}{\pi \cos \theta}, \]
with
\[ I_q(\theta) = \int_0^{+\infty} \int_0^{+\infty} x^q y^q \exp \left( -\frac{x^2 + y^2 - 2xy \sin \theta}{2\cos^2 \theta} \right) dxdy. \]

Under polar system, we have:
\[ I_q(\theta) = \int_0^{+\infty} \int_0^{\frac{\pi}{2}} r^{2q} (\cos \varphi \sin \varphi)^q \times \exp \left( -\frac{r^2}{2\cos^2 \theta} (1 - 2 \cos \varphi \sin \varphi \sin \theta) \right) rdrd\varphi. \]
\[ = \int_0^{\frac{\pi}{2}} \left( \cos \varphi \sin \varphi \right)^q \int_0^{+\infty} r^{2q+1} e^{-\alpha(\varphi)r^2} dr d\varphi, \]
where \( \alpha(\varphi) = \frac{1 - 2 \cos \varphi \sin \varphi \sin \theta}{2\cos^2 \theta} \geq 0 \). Using the definition of Gamma function,
\[ \int_0^{+\infty} r^{2q+1} e^{-\alpha r^2} dr = \alpha^{-q-1} \int_0^{+\infty} t^{2q+1} e^{-t^2} dt \]
\[ = \frac{\alpha^{-q-1}}{2} \Gamma(q+1). \]
We have,

\[ I_q(\theta) = \int_0^{\frac{\pi}{2}} (\cos \varphi \sin \varphi)^q \frac{\Gamma(q + 1)(2 \cos^2 \varphi)^{q+1}}{2(1 - 2 \cos \varphi \sin \varphi \sin \theta)^{q+1}} \]

\[ = 2^q (\cos \theta)^{2q+2} \Gamma(q + 1) \times \int_0^{\frac{\pi}{2}} \frac{(\cos \varphi \sin \varphi)^q}{(1 - 2 \cos \varphi \sin \varphi \sin \theta)^{q+1}} d\varphi \]

\[ = 2^q (\cos \theta)^{2q+2} \Gamma(q + 1) \times \int_0^{+\infty} \frac{t^q}{(1 + t^2 - 2t \sin \theta)^{q+1}} dt, \]

where we set \( t = \tan \varphi \). Now we need to compute

\[ A_q(\theta) := \int_0^{+\infty} \frac{t^q}{(1 + t^2 - 2t \sin \theta)^{q+1}} dt. \]

When \( q = 0, 1 \), we can get the explicit formula. Indeed, we have

\[ A_0(\theta) = \frac{1}{\cos \theta} \left( \frac{\pi}{2} + \theta \right) \]

and

\[ A_1(\theta) = \frac{1}{2 \cos^4 \theta} \left( \cos^2 \theta + \frac{\pi}{2} \sin \theta \cos \theta + \theta \sin \theta \cos \theta \right). \]

Thus we have

\[ I_0(\theta) = \cos \theta \left( \frac{\pi}{2} + \theta \right) \]

and

\[ I_1(\theta) = \cos^2 \theta + \frac{\pi}{2} \sin \theta \cos \theta + \theta \sin \theta \cos \theta. \]

Note that explicitly we have,

\[ E(|XY|^{0}) = \frac{I_0(\theta) + I_0(-\theta)}{\pi \cos \theta} = 1, \]

and

\[ E(|XY|^{1}) = \frac{I_1(\theta) + I_1(-\theta)}{\pi \cos \theta} = \frac{2}{\pi} (\cos \theta + \theta \sin \theta). \]
Then,
\[ A_q(0) = \int_0^{+\infty} t^q (1 + t^2)^{-q-1} dt = \frac{1}{2} \int_0^{+\infty} u^{q-1} (1 + u)^{-q-1} du = \frac{1}{2} B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right). \] (4.20)

Moreover, we have
\[ A_q'(\theta) = \int_0^{+\infty} 2t^q (q + 1) \cos \theta \frac{t^2 - 2t \sin \theta}{(1 + t^2)^{q+2}} dt, \]
and
\[ A_q''(\theta) = \int_0^{+\infty} 4t^q (q + 2) \cos^2 \theta \frac{t^2 - 2t \sin \theta}{(1 + t^2)^{q+3}} dt. \] (4.21)

Hence,
\[ A_q''(0) = 4(q + 1)(q + 2) \int_0^{+\infty} t^q \frac{t^2 - 2t \sin \theta}{(1 + t^2)^{q+3}} dt = 2(q + 1)(q + 2) B\left(\frac{q + 3}{2}, \frac{q + 3}{2}\right). \]

By Taylor series properties, we get an approximation
\[ E(|XY|^q) \approx I_q(\theta) + I_q(-\theta) \pi \cos \theta \approx \frac{2^{q+1}(\cos \theta)^{2q+1} \Gamma(q + 1)}{\pi} (A_q(0) + \frac{A_q''(0)}{2} \theta^2). \]

\[ \Box \]

Now we return to Lemma 8. The covariance matrix is
\[ \mathbb{E}(ZZ^T) = \begin{pmatrix} a^2 & ab \sin \theta \\ ab \sin \theta & b^2 \end{pmatrix}, \]

Then we have:
\[ \mathbb{E}(|XY|^q) \approx \frac{2^q (\cos \theta)^{2q+1} \Gamma(q + 1)}{\pi} |ab|^q (2A_q(0) + A_q''(0) \theta^2). \]

Now go back to the case where \( X = \partial_x \dot{U}(0) \) and \( Y = \partial_y \dot{U}(z) \). We thus have,
\[ \mathbb{E}(|XY|^q) \approx \frac{2^q \Gamma(q + 1) \alpha_x^q \alpha_y^q \omega_q}{\pi} \left( \frac{\Gamma_{xy}(z)}{\alpha_x \alpha_y} \right), \]
\[ \Box \]
where \( \tilde{\omega}_q(t) = (1 - t^2)^{2q+1} (2A_q(0) + A_q''(0) \arcsin(t)^2 + o(t^2)) \). Combining all terms, finally we get that:

\[
\mathbb{E}(TV_q(U))^2 = \frac{2q \Gamma(q + 1)}{\pi} \sum_{x \in \Omega} \alpha_x^{2q} \tilde{\omega}_q \left( \frac{\Gamma_{xx}(z)}{\alpha_x^2} \right) + 2\alpha_x \alpha_y^{q} \tilde{\omega}_q \left( \frac{\Gamma_{xy}(z)}{\alpha_x \alpha_y} \right) + \alpha_y^{2q} \tilde{\omega}_q \left( \frac{\Gamma_{yy}(z)}{\alpha_y^2} \right),
\]

where \( \tilde{\omega}_q(t) = (1 - t^2)^{2q+1} (2A_q(0) + A_q''(0) \arcsin(t)^2 + o(t^2)) \).

By (4.20) and (4.21), we have:

\[
\tilde{\omega}_q(t) = \left( B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right) + 2(q + 1)(q + 2)B\left(\frac{q + 3}{2}, \frac{q + 3}{2}\right) t^2 + o(t^2) \right) \times (1 - t^2)^{2q+1} = B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right) + 2(q + 1)(q + 2)B\left(\frac{q + 3}{2}, \frac{q + 3}{2}\right) t^2 + o(t^2) \\
= B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right) \times \left(1 + \frac{2q + 1}{2} t^2 + o(t^2)\right) = B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right) \left(1 + \frac{2q t^2}{2}\right) + o(t^2),
\]

which is consistent with the case \( q = 1 \) in [50]. Thus, when \( t < 1 \), we can use \( B\left(\frac{q + 1}{2}, \frac{q + 1}{2}\right) \left(1 + \frac{2q t^2}{2}\right) \) to approximate \( \tilde{\omega}_q(t) \). Recall that

\[
\mathbb{E}(TV_q(u * W)) = \frac{\Gamma\left(\frac{q + 1}{2}\right)}{2 \sqrt{\pi} |\Omega|} \left( \alpha_x^q + \alpha_y^q \right),
\]

where \( |\Omega| \) should be discarded as this is due to normalization. Put everything together, we should approximate the variation by

\[
\sigma_a^2 := \frac{2^{q-1} \Gamma^2\left(\frac{q + 1}{2}\right)}{q} \sum_{x \in \Omega} \frac{\|\Gamma_{xx}\|_2^2}{\alpha_x^{2q}} + 2 \cdot \frac{\|\Gamma_{xy}\|_2^2}{\alpha_x^{2q} \alpha_y^{2q}} + \frac{\|\Gamma_{yy}\|_2^2}{\alpha_y^{4-2q}}, \tag{4.22}
\]

which is again consistent with the previous result given by [50] when \( q = 1 \). Now we resume the calculation of SQ-Index into the following algorithm.
Algorithm 4 Algorithm: Computation of SQ Index of $u$

1: Calculate the derivatives $\partial_x u, \partial_y u$ and deduce their $l^q$ and $l^2$ norm, set $\alpha_x = \|\partial_x u\|_2$, $\alpha_y = \|\partial_y u\|_2$.

2: Calculate (in Fourier domain) the components of the auto-correlation gradient matrix $\Gamma$.

- Calculate the FFT $\hat{u}$ of $u$.
- Calculate the FFT of the derivatives by: $\|\hat{\partial}_x \hat{u}(\epsilon)\|_2^2 = 2 \sin^2\left(\frac{\pi \epsilon x}{M}\right)\|\hat{u}(\epsilon)\|_2^2$ and $\|\hat{\partial}_y \hat{u}(\epsilon)\|_2^2 = 2 \sin^2\left(\frac{\pi \epsilon y}{M}\right)\|\hat{u}(\epsilon)\|_2^2$.
- Calculate: $\|\hat{\Gamma}_{xx}\| = \|\hat{\partial}_x \hat{u}\|_2^2$, $\|\hat{\Gamma}_{xy}\| = \|\hat{\partial}_x \hat{u}\|\|\hat{\partial}_y \hat{u}\|$, $\|\hat{\Gamma}_{yy}\| = \|\hat{\partial}_y \hat{u}\|_2^2$.

3: Calculate $\mu$ and $\sigma_a$:

4: $\mu = \frac{\Gamma(\frac{q+1}{2})}{\sqrt{\pi}} \left(\frac{2}{\|\hat{u}\|}\right)^\frac{q}{2} (\alpha_x^q + \alpha_y^q)$;

5: $\sigma_a^2 := \frac{2^{q-1} \Gamma(\frac{q+1}{2})^2}{\pi} \left(\sum_{x \in \Omega} \frac{\|\hat{\Gamma}_{xx}\|_2^2}{\alpha_x^{2q}} + 2 \cdot \frac{\|\hat{\Gamma}_{xy}\|_2^2}{\alpha_x^{2q-2}\alpha_y^q} + \frac{\|\hat{\Gamma}_{yy}\|_2^2}{\alpha_y^{2q}}\right)$.

6: 

7: Calculate $S_q(u) = -\log_{10} \Psi\left(\frac{\mu - TV(u)}{\sigma_a}\right)$.

58
4.4 Application of SQ-Index for constrained TV model parameter selection

In this section, we will illustrate how to use SQ-Index for variational model parameter selection. We will introduce a constrained TV model, which is proved equivalent with the classical total variation model. The reason to apply SQ-Index for the convex model instead the classical total variation model is that we want to show SQ-Index can estimate the Gaussian noise level.

4.4.1 A constrained total variation model

The Total Variation(TV) model is the most famous and efficient variational models for image processing. The biggest advantage is that it can preserve well the edge information for restoring corrupted natural images. The image is supposed to be smooth inside the image and discontinuous on the boundaries. The BV space corresponds to the TV properties: the discontinuities along lines represent edges in natural images. Given a corrupted image \( u \in BV(\Omega) \), we give a constrained form of TV as:

\[
\min_{u \in D} TV(u),
\]

(4.23)

where

\[
D = \{ u : \|Au - g\|_2^2 \leq c^2 \}.
\]

and \( A \) can represent the blur kernel for image deblurring case. The constrained condition describes the limit of \( l_2 \)-norm of \( Au^* - g \). We can reformulate (4.23) with a Lagrange multiplier to a unconstrained problem:

\[
\min TV(u) + \lambda\|Au - g\|^2;
\]

(4.24)

which is the classical total variation model. Chambolle demonstrates in [18] that the two models are equivalent in the sense that for every parameter \( c \) in constrained model, there is a unique \( \lambda \) in unconstrained TV model.
4.4.2 Existence of the solution of the constrained model

Theorem 6. Assume that an image \( g \in \Omega \), blurred by a blur kernel \( A \) with a Gaussian noise \( n \). The solution of problem (4.23) exists.

Proof. Since \( u \in BV(\Omega) \), we note \( \Omega' := \Omega \cap \{ u \| Au - g \| \leq c \} \). Then \( \Omega' \) is a bounded open subset of \( \Omega \) with Lipschitz boundary. For \( u \in \Omega' \) we take a minimising sequence \( \{ u_n \} \) which ensures that \( TV(u_n) \) is bounded. The boundedness of \( TV(u_n) \) means that \( \{ u_n \} \) is a bounded sequence in the space \( BV(\Omega') \). Applying the Poincare inequality in [9], we get:

\[
\| u_n - m_{\Omega'}(u_n) \|_2 \leq C \int_{\Omega'} |D(u_n - m_{\Omega'}(u_n))| = C \int_{\Omega'} |Du_n|
\]

where \( m_{\Omega'}(u_n) = \frac{1}{|\Omega'|} \int_{\Omega'} u_n \, dx \), \( C \) is a constant. Since \( \Omega' \) is closed, \( \| u_n - m_{\Omega'}(u_n) \|_2 \) is bounded for each \( n \). Then \( \| A(u_n - m_{\Omega'}(u_n)) \|_2 \) is bounded in \( L^2(\Omega') \) because of the continuity and boundedness of \( A \). In addition, \( \| Au_n - g \|_2 \) is in a closed convex set thus bounded in \( L^2(\Omega') \) and \( L^1(\Omega') \), which results in:

\[
|m_{\Omega'}(u_n)| \times \| A \|_1 = \| A(u_n - m_{\Omega'}(u_n)) - Au_n \|_1 \leq \| A(u_n - m_{\Omega'}(u_n)) \|_1 + \| Au_n \|_1.
\]

Then \( m_{\Omega'}(u_n) \) is uniformly bounded. As we know \( \{ u_n - m_{\Omega'}(u_n) \} \) is bounded, thus, by compactness of \( L^1(\Omega') \) in \( BV \), there exists a subsequence \( \{ u_{n_k} \} \) which converges to \( u \) in \( L^1(\Omega') \). By the lower semi-continuity of the BV norm, \( TV(u) \leq \liminf TV(u_{n_k}) \) in \( L^1(\Omega') \). Since the operator \( A \) is continuous, \( Au_{n_k} \) also converges weakly to \( Au \) in \( L^2(\Omega') \). Based on Fatou’s lemma, \( u \) minimizes \( TV(u) \) in the constrained condition. Then we prove the existence of problem (4.23).

4.4.3 Numerical algorithm for constrained TV model

Many methods can be applied to solve the constrained model. People always transform it to the unconstrained one (4.24) and then many methods can be used such as split-bregman method [17, 35, 53], gradient based method [34, 48, 92, 93], primal-dual method [20, 29, 39, 93]. Our contribution here is to use primal-dual-projection
method to solve the constrained model directly. In order to solve the constrained model numerically, we make a slight change for the constrained condition as follows:

$$\min_{u \in D} TV(u)$$  \hspace{1cm} (4.25)

where

$$D = \{ u : \| Au - g \|^2_2 = c^2 \}.$$  

We can easily prove that the model (4.25) is equivalent with model (4.23) with mild condition.

**Lemma 10.** If $u$ is not constant, the solution $u$ of model (4.25) must be on the surface of $\| Au - g \|^2_2 = c^2$.

**Proof.** Suppose that $\| Au - g \|^2_2 < c^2$, then the solution of model (4.25) is the same as the solution of unconstrained problem $\min TV(u)$ since the constrained condition is inactive. Then, the solution $u^*$ of $\min TV(u)$ must be a constant for every pixel of $u$ in order to make $TV(u) = 0$. Obviously, the constant solution is not the well recovered image. Then we deduce that the solution of (4.25) must be on the surface of $\| Au - g \|^2_2 = c^2$ if solution $u$ is not constant. 

Since problem (4.25) is equivalent with problem (4.23), we reformulate this problem as:

$$\min_{u, Au = v, \| v - g \| \leq c} TV(u).$$  \hspace{1cm} (4.26)

The primal-dual form of problem (4.26) is:

$$\max_{\| p \| \leq 1, q : \| v - g \|_2 \leq c} \min_{u, v} -\langle u, \text{div} \, p \rangle + \langle Au - v, q \rangle + \chi(p),$$  \hspace{1cm} (4.27)

where $\chi(p)$ is the characteristic function.

Then the numerical solution is iterated by optimizing:

$$\hat{p}^{k+1} = \arg \max_p \{ \langle \nabla u^k, p \rangle - \frac{1}{2\tau} \| p - p^k \|^2 + \chi(p) \};$$  \hspace{1cm} (4.28)

$$q^{k+1} = \arg \max_q \{ \langle Au^k - v^k, q \rangle - \frac{1}{2\tau} \| q - q^k \|^2 \};$$  \hspace{1cm} (4.29)
\[
\begin{align*}
    u_{k+1} &= \arg \min_u \left\{ -\langle u, \text{div} p^{k+1} \rangle + \langle Au, q^{k+1} \rangle + \frac{1}{2\tau} \| u - u^k \|^2 \right\}; \hspace{1cm} (4.30) \\
    \tilde{v}_{k+1} &= \arg \min_v \left\{ \langle -v, q^{k+1} \rangle + \frac{1}{2\tau} \| v - v^k \|^2 \right\}; \hspace{1cm} (4.31) \\
    v^{k+1} &= \text{P}(\tilde{v}^{k+1}), \hspace{1cm} (4.32)
\end{align*}
\]

where \( \text{P} \) is the projection of \( \tilde{v}^{k+1} \) onto \( \| v - g \|_2 = c \).

The solution is:

\[
\begin{align*}
    p^{k+1} &= p^k + \tau \ast \nabla u^k, \\
    \tilde{p}_{ij}^{k+1} &= \frac{p_{ij}}{\max(1, |p_{ij}|)}, \hspace{1cm} (4.33) \\
    q^{k+1} &= q^k + \tau(Au^k - v^k); \hspace{1cm} (4.34) \\
    u^{k+1} &= u^k + \tau(\text{div} p^{k+1} - A^T q^{k+1}); \hspace{1cm} (4.35) \\
    \tilde{v} &= v^k + \tau q^{k+1}.
\end{align*}
\]

To calculate \( v^{k+1} \), first we calculate \( \| \tilde{v} - g \|_2 \). If \( \| \tilde{v} - g \|_2 \leq c \), then the projection of \( \tilde{v} \) is itself. Then \( v^{k+1} = \tilde{v} \). If \( \| \tilde{v} - g \|_2 > c \), then we have:

\[
    v^{k+1} - g = c \cdot \frac{\tilde{v} - g}{\| \tilde{v} - g \|}
\]

Then

\[
    v^{k+1} = \frac{c}{\| \tilde{v} - g \|} \tilde{v} + (1 - \frac{c}{\| \tilde{v} - g \|})g.
\]

To put the two cases together, we have:

\[
    v^{k+1} = \theta \tilde{v} + (1 - \theta)g, \hspace{1cm} (4.36)
\]

where \( \theta = \frac{c}{\max(c, \| \tilde{v} - g \|)} \). Now we resume the above method into the following algorithm:
Algorithm 5 Algorithm for solving constrained TV model.

1: Initialize: $p^{(0)} = 0, q^{(0)} = 0, u^{(0)} = g, v^{(0)} = g.$
2: Do $k = 0, 1, \ldots, \text{until} \|u^{k+1} - u^k\| / \|u^k\| < \epsilon$
   • Compute $p^{k+1}$ by (4.33),
   • Compute $q^{k+1}$ by (4.34),
   • Compute $u^{k+1}$ by (4.35),
   • Compute $v^{k+1}$ by (4.36).
3: Output $u$.

4.4.4 SQ-Index for selection of parameter $c$ in constrained TV model

Before applying our new metric SQ-Index to select parameter $c$ in the constrained model, let us first give an introduction of the state-of-the-art methods for regularization parameter selection. A general review of parameter selection methods can be found in [45]. The most popular ones are the generalized cross validation (GCV) [33, 36, 51], L-curve method [38], the discrepancy principle [13, 57, 88], and the variational Bayes approaches [61]. GCV method requires to solve a minimizer to get the solution of optimal parameter. However, the regularization term needs to be quadratic. The algorithm for solving the minimizer is always complicated and sometimes gives a undersmoothing solution. L-curve method is also efficient for parameter selection. It is by comparing the value of $\log TV(f(\lambda))$ and $\log \|Hf(\lambda) - g\|_2^2$. The disadvantage of L-curve method is that it is time-consuming. Moreover, the solution given by it is usually a little bit over-smoothing. Another important and efficient method is based on discrepancy principle. By selecting $\lambda$ which matches the residual norm to an upper bound, a good solution should be in the set:

$$D = \{u : \|Au - g\|_2^2 \leq c^2\}$$
where $c$ is the constant which is equivalent to the noise standard deviation. In Y. Wen’s paper [88], a discrepancy principle based on primal-dual model is proposed for Gaussian noise removal. The parameter $\lambda$ is determined by a constrained condition $\|Au-g\| = c$. Now we want to choose the constrained parameter $c$ with our SQ-Index. In order to be clear, we illustrate the constrained TV model again:

$$\min_{u \in D} \text{TV}(u)$$

where

$$D = \{u : \|Au - g\|^2 \leq c^2\}.$$  

and $c$ is a constant depending on the noise level. Chambolle has proved that the constrained model (4.23) is equivalent with the typical unconstrained TV model in the sense that for every $c$ in model (4.23) between 0 and $\|g - \langle g \rangle\|$ where $\langle g \rangle$ is the average pixel of $g$, there exists an unique $\lambda$ in unconstrained model correspondingly. We can refer to Lemma 4.1 in [19].

In order to select parameter $c$ in Algorithm 5, we need to iterate $c$ with a scope and solve the problem with each possible $c$. In order to select faster, we use two iterations. First we set $c$ from 1 to $N$ where $N = g - \langle g \rangle$ which is all possible range of parameter $c$. The iteration step size can be set as 0.1 for our constrained TV model. The algorithm can be summarize in the following algorithm. We note that the step size can be changed based on the scope $N$ and different models.

\begin{algorithm}
\caption{Algorithm for choosing $c$ with SQ-Index}
\begin{algorithmic}[1]
\FOR {$i = 1$ to $N$}
\STATE solve model with Algorithm 5 under $\|Au - g\| \leq i$ and get solution $u$.
\STATE calculate SQ-Index with solution $u$.
\ENDFOR
\STATE Select parameter $i$ which produces the biggest SQ-Index.
\end{algorithmic}
\end{algorithm}
4.5 Numerical results

In this section, we present the performance of image denoising and image deblurring with parameter selected by SQ-Index. We use PSNR and ISNR to measure the performance. PSNR is defined in Chapter 2 and ISNR is defined as:

$$
\text{ISNR} = 10 \log_{10} \frac{\|g - I\|_2^2}{\|u - I\|_2^2}
$$

where $g, u, I$ are respectively the observed image, the restored image and the original image. In experiment 1, we show the sensitivity of $q$ for parameter selection and the performance of image restored based on the selected parameter. In experiment 2, we show that we can estimate Gaussian noise level by using SQ-Index. Furthermore, we illustrate with the fixed $q$, what the curve of SQ-Index is like for different parameter $c$. In Experiment 3, we compare our method with four existed methods. We restore image with Gaussian blur and average blur respectively with 20, 30 40 BSNR Gaussian noise. BSNR is defined as:

$$
\text{BSNR} = 10 \log_{10} \frac{\|g\|_2^2}{\|n\|_2^2}.
$$

4.5.1 Selection of $q$ for SQ-Index

Now we want to discuss the issue: how to select $q$ for SQ-Index? We know that when $q = 1$, SQ-Index is the same with S-Index proposed by [49]. Thus, why we want to change $q$? Which $q$ can help choose a better regularization parameter in order to get a higher performance for image restoration? Now we use our experiment to demonstrate. For the moment, we do not have a strict mathematical method to prove which $q$ is optimal. For every image, we blur it with a Gaussian blur kernel of size $9 \times 9$ with variance 3. Then we add Gaussian noise with level 1 and 5 respectively for the four images. For each $q$, we use the above constrained primal-dual algorithm to restore images with different parameter $c$. We select the parameter which gives the biggest SQ-Index. Then we iterate with different values of $q$ to calculate SQ-Index.
Then we can get the sensitivity of $q$ with regard to the corresponding PSNR obtained by the restored image with the parameter selected by SQ-Index. The result is shown in Figure 4.2.

![Diagram showing PSNR values for different images with Gaussian blur and different noise levels.](image)

Figure 4.2: PSNR value of restored image with parameters selected with different $q$ by SQ-Index for image with Gaussian blur.

We observe that the PSNR value is relatively better with $q$ between 0.9 and 1 for restoring image with Gaussian blur with additive Gaussian noise. We also test the sensibility of $q$ for SQ-Index for restoring image with motion blur and additive Gaussian noise with level 2 and 5. The sensitivity of $q$ is given in Figure 4.3.
We observe that the PSNR value is relatively better with $q$ between 0.95 and 1.1 for restoring image with motion blur and Gaussian noise.

4.5.2 Estimation of Gaussian noise level with SQ-Index

With constrained model (4.23), we use SQ-Index to select the parameter $c$ in this model. Parameter $c$ is iterated from 0.1 to standard deviation of observed image with step size 0.1 each time. In order to get the optimized parameter faster, we can refer to Algorithm 6. By the primal-dual algorithm, we find that the selected parameter $c$
Table 4.1: Noise level estimation with SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>Noise level</th>
<th>Noise level estimated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>2</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>house</td>
<td>2</td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>parrot</td>
<td>2</td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

is quite close with the noise standard deviation for the observed image.

Table 4.1 gives more results about the estimation of noise level with SQ-Index for different images with different noise level. We set $q = 0.95$ for the experiment.

The curve of SQ-Index with different parameter $c$ in the constrained model is not necessarily a convex smoothing curve. See Figure 4.4. Different $q$ can bring similar curves. However, the maximum value of SQ-Index can correspond to different parameter $c$ under different $q$. We try to use $q = 1$ to select parameters for different models but find that it can not adapt to different models in the sense that by $q = 1$, the selected parameter can not always bring a good performance of image restoration. Thus, to find a good $q$ for different model, we can get a better restoration performance for different models. Compared with using S-Index, using SQ-Index can help us choose parameters more precisely. Please see next chapter for more examples. The $q$ is decided by training some known images in advance for different models. We will illustrate it further in the next chapter.
Figure 4.4: SQ Index curve for constrained TV model for image (a) Parrot with 40 dB Gaussian noise; (b) Cameraman with 20 dB Gaussian noise.

### 4.5.3 Image denoising performance

With constrained model (4.23), we set $A = I$, then we can use this model for image denoising. The tested images are degraded by Gaussian noise with standard deviation $\sigma = 10, 20$ respectively. We compare our method with Wen Z.’s discrepancy principle method [88], here we name it DP method for short. We set Wen’s parameter $c = 0.99$ which is the one the author uses in their paper and set $q = 0.95$ for our method. We implement all experiments in MATLAB R2013a on a personal computer with 4.0G
memory. The stopping criteria for the algorithm of solving model (4.23) is:

\[
\frac{\|u_{k+1} - u_k\|}{\|u_k\|} < 10^{-4}.
\]

The performance of image denoising with parameter selected by SQ-Index is in Figure 4.5 and 4.6.

![Figure 4.5](image1.png)

Figure 4.5: (a) Cameraman with noise level 10; (b) DP method: ISNR=65.8dB; (c) SQ-Index: ISNR=65.8dB.

![Figure 4.6](image2.png)

Figure 4.6: (a) Boat with noise level 20; (b) DP method: ISNR=65.1dB; (c) SQ-Index: ISNR=65.1dB.

From the denoising performance visually and the ISNR value, we see obviously that our method and Wen Z.’s method are both very effective in image denoising for automatically selecting parameters without information of the noise level.

### 4.5.4 Image deblurring performance

In this subsection, we show the performance of image deblurring under different Gaussian noise level. We corrupt images with two blur kernels. One is Gaussian blur
with a window size $9 \times 9$ with standard deviation 3, the other is average blur with a window size 9. Then we add Gaussian noise for the tested images by three noise levels with BSNR=20dB, 30dB, 40dB respectively. We compare our performance with [14,52,65,88]. In [14], S.D. Babacan, R. Molina restored image by Bayesian inference and variational approximation which is called BMA method for short. For more information, we can refer to [8,73]. Jose M. Bioucas-Dias proposed a majorization-minimization approach to choose regularization parameter for TV models based on Bayesian approach under a Jeffreys’ prior. We call this method BFO for short. In [52], the author uses the primal variable and apply GCV to parameter selection in each step. We call this method GCV for short. In [88], the author uses discrepancy principle to solve the parameter selection with a pre-defined parameter $\tau$. We call this method DP for short. This method also uses the constrained model which is quite similar with our model. Now we compare our method with the other four methods for image deblurring performance for image 'Cameraman', 'Lena' and 'Parrot'. The performance is in Table 4.2, 4.3. Noted that for the other three methods, there is no performance for 'Parrot' in their original paper. Based on the sensitivity analysis, we set $q = 0.95$ for deblurring images with Gaussian blur, $q = 1.03$ for deblurring images with motion blur.

We can see from the tables that with SQ-Index, the performance of image deblurring and denoising simultaneously outperforms the other parameter selection methods on average. The solution of other methods are not explicit which depend heavily on the restoration model. Compared with these methods, SQ-Index is very straightforward to apply and it can be used to different models with a very explicit way. Thus, SQ-Index is more convenient to be adapted to different models.
Table 4.2: Performance of image denoising and deblurring under Gaussian blur (ISNR).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cameraman</td>
<td>20dB</td>
<td>1.72</td>
<td>2.21</td>
<td>1.82</td>
<td>2.59</td>
<td><strong>2.60</strong></td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>2.63</td>
<td>3.59</td>
<td>3.43</td>
<td>4.10</td>
<td><strong>4.17</strong></td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td>3.39</td>
<td>5.78</td>
<td>5.02</td>
<td>6.00</td>
<td><strong>6.04</strong></td>
</tr>
<tr>
<td>lena</td>
<td>20dB</td>
<td>2.87</td>
<td>2.99</td>
<td>2.57</td>
<td>3.12</td>
<td>3.07</td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>3.87</td>
<td>3.82</td>
<td>4.17</td>
<td>3.98</td>
<td><strong>4.20</strong></td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td>4.78</td>
<td>4.41</td>
<td>5.44</td>
<td>5.41</td>
<td><strong>5.67</strong></td>
</tr>
<tr>
<td>parrot</td>
<td>20dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>3.21</td>
<td>3.07</td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4.85</td>
<td><strong>4.93</strong></td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>6.56</td>
<td><strong>6.77</strong></td>
</tr>
</tbody>
</table>

Table 4.3: Performance of image denoising and deblurring under average blur (ISNR).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cameraman</td>
<td>20dB</td>
<td>2.42</td>
<td>3.27</td>
<td>2.88</td>
<td>3.87</td>
<td><strong>3.90</strong></td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>5.41</td>
<td>5.69</td>
<td>5.57</td>
<td>5.70</td>
<td><strong>5.93</strong></td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td><strong>8.57</strong></td>
<td>8.46</td>
<td>8.36</td>
<td>8.36</td>
<td>8.39</td>
</tr>
<tr>
<td>lena</td>
<td>20dB</td>
<td>3.72</td>
<td>4.05</td>
<td>3.15</td>
<td>3.79</td>
<td><strong>3.96</strong></td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>5.41</td>
<td><strong>5.69</strong></td>
<td>5.57</td>
<td>5.13</td>
<td>5.53</td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td><strong>8.42</strong></td>
<td>6.22</td>
<td>6.92</td>
<td>6.98</td>
<td>6.99</td>
</tr>
<tr>
<td>parrot</td>
<td>20dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4.11</td>
<td>4.01</td>
</tr>
<tr>
<td></td>
<td>30dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>6.14</td>
<td><strong>6.38</strong></td>
</tr>
<tr>
<td></td>
<td>40dB</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>8.56</td>
<td><strong>9.39</strong></td>
</tr>
</tbody>
</table>
4.6 Conclusion

In this chapter, we propose a new Sharpness metric called SQ-Index. We show the definition and properties of SQ-Index. We apply it to a constrained TV model. SQ-Index can help choose regularization parameter automatically. Compared with other parameter selection methods, SQ-Index can give a very competitive performance for restoring images. Moreover, it is very easy to apply for different models. We will show the robustness of SQ-Index in next chapter.
Chapter 5

More applications for SQ-Index

In order to see the robustness of SQ-Index for different models parameter selection, we give more examples in this chapter. We will give the performance of SQ-Index for parameter selection for several models including TV unconstrained model, Poisson variational model, Rician noise removal MAP model. We illustrate the effectiveness of SQ-Index for each model for both denoising and deblurring.

5.1 SQ-Index for Total Variation model parameter selection

5.1.1 SQ-Index for image denoising with Gaussian noise

Since we have introduced TV model in continuous and discrete form in the above chapters. Here we give the discrete form directly:

\[ TV(u) + \lambda \| u - g \|_2^2. \]  

(5.1)

Now we will use SQ-Index to select the parameter \( \lambda \). We consider separately the denoising model and deblurring model.

The denoising algorithm is by Chambolle’s projection method [18]. We summarize the algorithm in Algorithm 7. We iterate parameter \( \lambda \) to get a series of solution \( u \). We choose the parameter \( \lambda \) which brings the biggest SQ-Index for different \( u \) corresponded. We set \( q = 1.05 \) for TV denoising experiments.

We summarize the performance in Figure 5.1, 5.2 and in Table 5.1. In Table 5.1, The column 'PSNR' is the best PSNR we can get by selecting the best PSNR value
Algorithm 7 Chambolle’s projection algorithm for image denoising

1: Initialize: \( p^{(0)} = 0, u^{(0)} = g, \tau = 0.01, \epsilon = 10^{-4} \).

2: Do \( k = 0,1,\ldots,\) until \( \| p^{k+1} - p^k \|/\| p^k \| < \epsilon \):

3: \( p_{i,j}^{k+1} = p_{i,j}^k + \tau (\nabla (\text{div} p^k - g/\lambda))_{i,j} / (1 + \tau (\| \nabla (\text{div} p^k - g/\lambda) \|)_{i,j}^2) \).

4: \( u = g - \lambda \text{div} p_{i,j}^{k+1} \)

for each \( \lambda \). The column ‘PSNR(SQ)’ represents the PSNR of restored image with parameter selected by SQ-Index. The column \( \lambda \) is the parameter which corresponds to the maximum PSNR value while the column \( \lambda (SQ) \) is the parameter selected by SQ-Index. The last column ‘RE of PSNR’ represents ‘relative error of PSNR’ between best PSNR and PSNR selected by SQ-Index.

Table 5.1: Performance of Gaussian noise removal with parameter selected by SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>Noise Level</th>
<th>PSNR</th>
<th>PSNR(SQ)</th>
<th>( \lambda )</th>
<th>( \lambda ) (SQ)</th>
<th>RE of PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>20</td>
<td>28.82</td>
<td>28.15</td>
<td>14</td>
<td>20</td>
<td>2.32%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>26.96</td>
<td>26.89</td>
<td>23</td>
<td>21</td>
<td>0.26%</td>
</tr>
<tr>
<td>Parrot</td>
<td>20</td>
<td>28.84</td>
<td>28.01</td>
<td>14</td>
<td>22</td>
<td>2.88%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>26.78</td>
<td>26.31</td>
<td>23</td>
<td>32</td>
<td>1.75%</td>
</tr>
<tr>
<td>Hill</td>
<td>20</td>
<td>29.48</td>
<td>29.03</td>
<td>15</td>
<td>11</td>
<td>1.52%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>27.94</td>
<td>27.64</td>
<td>24</td>
<td>20</td>
<td>1.07%</td>
</tr>
<tr>
<td>Couple</td>
<td>20</td>
<td>28.86</td>
<td>28.21</td>
<td>14</td>
<td>20</td>
<td>2.25%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>27.05</td>
<td>27.03</td>
<td>23</td>
<td>24</td>
<td>0.04%</td>
</tr>
</tbody>
</table>
Figure 5.1: Column 1: image corrupted with Gaussian noise with noise intensity 20; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: the sensitivity of $q$ with regard to PSNR of restored image.
Figure 5.2: Column 1: image corrupted with Gaussian noise with noise intensity 30; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: the sensitivity of $q$ with regard to PSNR of restored image.

From the results of image restoration for Gaussian noise removal, we observe that SQ-Index can give a very good performance with a well selected parameter. This can be seen from the tiny percentage of the relative error with best PSNR. From the figures in the last column, we can observe the sensitivity of $q$. For different image, the best performance corresponds with different $q$. Therefore, we try to choose a $q$ which is relatively good for all testing images. The sensitivity of $q$ depends more on
image itself than the different noise levels since for different noise level, the curves of the sensitivity of \( q \) for identical image are similar.

### 5.1.2 SQ-Index for image deblurring with Gaussian noise

Now we also try to recover blurred image with TV unconstrained model. The model is:

\[
\min_{u \in \Omega} |\nabla u| + \lambda ||Au - g||^2_2
\]  

(5.2)

where \( A \) is the blur kernel. Since the model is convex, we can transfer the min problem to a min-max problem as:

\[
\min_{u} \max_{p, ||p||_{\infty} \leq 1} \langle u, -\text{div} p \rangle + \frac{\lambda}{2} ||Au - g||^2_2 + \chi_p(P) - \frac{1}{2\beta} ||p - p^k||^2 + \frac{1}{2\tau} ||u - u^k||.
\]  

(5.3)

The algorithm to solve model (5.2) is in Algorithm 8.

**Algorithm 8** Algorithm for image deblurring with total variational model.

1: Initialize: \( p^{(0)} = 0, u^{(0)} = g, \beta = 0.01, \tau = 0.01, \epsilon = 10^{-4} \).

2: Do k = 0,1,...,until \( \frac{||u^{k+1} - u^k||}{||u^k||} < \epsilon \).

3: \( \tilde{p}^{k+1} = p^k + \beta \nabla u \),

4: \( p^{k+1} = \max(1, ||\tilde{p}^{k+1}||) \);

5: \( u^{k+1} = \frac{u^k + \tau \text{div} p^{k+1} + \lambda F^T s_q}{1 + \lambda F^T s_q} \).

Here we test two kernels including Gaussian kernel with 9*9 and a variance of 3 and average kernel with size 9. Then we add 10 additive Gaussian noise respectively. The performance of deblurring with the parameter selected automatically by SQ-Index is shown in Figure 5.3, 5.4 and Table 5.2. Here we use \( q = 0.95 \) for calculating SQ-Index for Gaussian blur and average blur.
Table 5.2: Performance of Gaussian noise removal under blurry condition with parameter selected by SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>BlurType</th>
<th>PSNR</th>
<th>PSNR(SQ)</th>
<th>(\lambda)</th>
<th>(\lambda(SQ))</th>
<th>RE of PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>Gaussian</td>
<td>23.11</td>
<td>22.97</td>
<td>2</td>
<td>1</td>
<td>0.61%</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>23.09</td>
<td>23.09</td>
<td>1.6</td>
<td>1.3</td>
<td>0%</td>
</tr>
<tr>
<td>Parrot</td>
<td>Gaussian</td>
<td>22.79</td>
<td>22.76</td>
<td>1.9</td>
<td>1.5</td>
<td>0.13%</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>22.13</td>
<td>22.11</td>
<td>1.7</td>
<td>1.5</td>
<td>0.09%</td>
</tr>
<tr>
<td>Boat</td>
<td>Gaussian</td>
<td>25.40</td>
<td>25.19</td>
<td>1.5</td>
<td>0.5</td>
<td>0.82%</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>25.21</td>
<td>25.06</td>
<td>1.5</td>
<td>1.1</td>
<td>0.31%</td>
</tr>
<tr>
<td>Pepper</td>
<td>Gaussian</td>
<td>23.69</td>
<td>23.68</td>
<td>1.6</td>
<td>1.8</td>
<td>0.02%</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>24.08</td>
<td>24.06</td>
<td>1.6</td>
<td>1.4</td>
<td>0.08%</td>
</tr>
<tr>
<td>Fingerprint</td>
<td>Gaussian</td>
<td>23.62</td>
<td>20.80</td>
<td>15</td>
<td>2</td>
<td>11.9%</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>21.83</td>
<td>19.90</td>
<td>17</td>
<td>4</td>
<td>8.8%</td>
</tr>
</tbody>
</table>
Figure 5.3: Column 1: image corrupted with Gaussian noise with noise intensity 10 under Gaussian blur with size 9 \times 9 and variance 3; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; column 4: sensitivity of $q$ for SQ-Index with respect to PSNR of restored image.
Figure 5.4: Column 1: image corrupted with Gaussian noise with noise intensity 10 under average blur with size 9; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index; Column 4: sensitivity of $q$ for SQ-Index with respect to PSNR of restored image.

We observe the deblurring performance with SQ-Index, most of images can be recovered very well. The performance is good compared with the best possible PSNR.
However, the texture image Fingerprint cannot be well restored since the theory of SQ-Index does not adapt to texture image. Because the edge information in texture image is not in phase information but magnitude information. Thus, based on phase information, SQ-Index does not work well for texture image.

5.2 SQ-Index for Poisson noise removal model parameter selection

In this section, we apply SQ-Index to select parameter for a variational model for Poisson noise removal. Poisson noise always appears in optical imaging, like astronomy images. A random variable $n$ with a Poisson distribution is:

$$p(n) = \frac{e^{-\lambda} \lambda^n}{n!}, \quad n = 0, 1, 2...$$  \hspace{1cm} (5.4)

where $\lambda$ represents the density of the distribution. Physically, $\lambda$ represents the expected value of the counts of the incident particles.

5.2.1 SQ-Index for image denoising with Poisson noise

Many image restoration methods have been developed based on Poisson noise statistical property. A maximum likelihood approach is proposed in [72]. M Bertero has developed a discrepancy principle for Poisson data to simultaneously denoise and deblur image [10]. In [?], the author uses a variational model to reconstruct image degraded by Poisson noise. The model is deduced from Bayes’s properties. We want to obtain the denoised image $u$ under the prior information of corrupted image $f$. Based on Bayesian statistics, we need to maximize $P(f|u)P(u)$. Let each $x \in \Omega$ have a Poisson noise distribution and assume the pixels $x_i$ in $f$ are independent. Then we have:

$$P(f(x)|u) = P_{u(x)}(f(x)) = \prod_i \frac{e^{-u(x_i)u(x_i)f(x_i)}}{f(x_i)!}.$$  \hspace{1cm} (5.5)
The TV term $P(u)$ depends on our prior information:

$$P(u) = e^{-\beta f_\alpha |\nabla u|},$$  \hspace{1cm} (5.6)

where $\beta$ is the regularization parameter. Then to maximize $P(f|u)P(u)$, we can equally minimize $-\log(P(f|u)P(u))$ as:

$$E(u) := \sum_i (u(x_i) - f(x_i) \log(u(x_i)) + \beta \int_\Omega |\nabla u|,$$  \hspace{1cm} (5.7)

where $u \in BV(\Omega)$ such that $\log u \in L^1(\Omega)$. Here we briefly give the numerical algorithm for this model. Since the model is convex, we can use primal-dual method to solve it. The original problem:

$$\min_u \hat{E}(u) = |\nabla u| + \beta \langle u - f \log u, 1 \rangle$$  \hspace{1cm} (5.8)

is equivalent with:

$$\min_u \max_{p, \|p\|\leq 1} \hat{E}(u, p) = \langle \nabla u, p \rangle + \beta \langle u - f \log u, 1 \rangle - \frac{1}{2\tau} \|p - p^k\|^2 + \frac{1}{2\tau} \|u - u^k\|^2.$$  \hspace{1cm} (5.9)

Then we have two subproblems:

$$p^{k+1} = \arg \max_{p \in P} \langle \nabla u^k, p \rangle - \frac{1}{2\tau} \|p - p^k\|_2^2; \hspace{1cm} (5.10)$$

$$\bar{u}^{k+1} = \arg \min -\langle \text{div} p^{k+1}, u \rangle + \beta \langle u - f \log u, 1 \rangle + \frac{1}{2\tau} \|u - u^k\|_2^2.$$  \hspace{1cm} (5.11)

The solution of (5.10) and (5.11) are:

$$\tilde{p}^{k+1} = p^k + \tau \nabla u^k,$$  \hspace{1cm} (5.12)

$$p^{k+1} = \frac{p^{k+1}}{\max(1, |\tilde{p}^{k+1}|)};$$

$$u^{k+1} = u^k + (\beta \tau - \text{div} p^{k+1} - u^k)u - \tau \beta f = 0,$$  \hspace{1cm} (5.14)

where $u^{k+1}$ is the solution of equation (5.14). We summarize the algorithm in Algorithm 9.

Now we iterate this algorithm with different parameter $\beta$ from 1 : 0.2 : 9.8 which needs to iterate 50 times. We calculate SQ-Index for different $\beta$ with $q = 1.15$. We get
Algorithm 9 Algorithm for solving variational model for Poisson denoising.

1: initiate $u^0 = f$, $p^0 = 0$, $\tau = 0.05$, $tol_{in} = 10^{-4}$.

2: for $m = 1$ to maxNum do
3:   solve $p^{k+1}$ by (5.12) and (5.13);
4:   solve $u^{k+1}$ by (5.14).
5:   If $\|u_k - u_{k+1}\|_2 / \|u_k\|_2 < tol_{in}$, $u^k = u^{k+1}$, break.
6: end for

a matrix of SQ-Index which has 50 different values. We choose the biggest value. The $\beta$ corresponded to the biggest SQ-Index is the parameter we choose finally. Now we show the performance of Poisson denoising by automatically selecting regularization parameter with SQ-Index. Poisson noise intensity is set by 8. We give also the sensitivity of $q$ with respect to PSNR of restored image. Particularly, we compare the PSNR of restored image by SQ-Index with $q = 1$ in order to illustrate that SQ-Index is a better metric to select parameter than S-Index for Poisson denoising model. The performance is listed in Figure 5.5 and Table 5.3. In Table 5.3, the column 'PSNR' is still the best PSNR by choosing the the PSNR value for each parameter. The column 'PSNR($q = 1$)' is the PSNR of restored image by SQ-Index with $q = 1$. The column 'PSNR($q = 1.15$)' is the PSNR of restored image by SQ-Index with $q = 1.15$. Column 'beta' is the parameter which brings the biggest PSNR while column 'beta($q = 1.15$)' is the parameter selected by SQ-Index with $q = 1.15$.

Table 5.3: Performance of Poisson noise removal with parameter selected by SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>PSNR</th>
<th>PSNR($q=1$)</th>
<th>PSNR($q=1.15$)</th>
<th>beta</th>
<th>beta($q=1.15$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>26.25</td>
<td>24.84</td>
<td>25.63</td>
<td>5.6</td>
<td>5.2</td>
</tr>
<tr>
<td>Parrot</td>
<td>25.76</td>
<td>22.79</td>
<td>24.13</td>
<td>5.2</td>
<td>6.8</td>
</tr>
<tr>
<td>Fluocells</td>
<td>29.30</td>
<td>26.04</td>
<td>28.00</td>
<td>4</td>
<td>6.4</td>
</tr>
<tr>
<td>Pepper256</td>
<td>26.86</td>
<td>25.11</td>
<td>24.44</td>
<td>5.2</td>
<td>6.7</td>
</tr>
<tr>
<td>House</td>
<td>27.24</td>
<td>25.92</td>
<td>25.96</td>
<td>5.2</td>
<td>7.8</td>
</tr>
<tr>
<td>Hill</td>
<td>26.84</td>
<td>26.04</td>
<td>26.23</td>
<td>5</td>
<td>5.8</td>
</tr>
</tbody>
</table>
Figure 5.5: Column 1: image restored with best PSNR-value (try and trial) which is corrupted by Poisson noise with noise intensity 8; Column 2: image restored with model parameter selected by SQ-Index for $q = 1$; Column 3: image restored with model parameter selected by SQ-Index for $q = 1.15$; Column 4: sensitivity of $q$ with respect to PSNR of restored image.
From Table 5.3, we can observe that with parameter $\beta$ selected by SQ-Index with $q = 1.15$, we can get very good performance of Poisson denoising. It is quite approximated with the best PSNR by manually try and trial. And on average, it outperforms the performance of image restoration by SQ-Index with $q = 1$. The sensitivity of $q$ for different image is quite different. We choose a $q$ which is relatively good for all tested images. Thus, we conclude that SQ-Index is very efficient for parameter selection for Poisson denoising variational model by choosing a good $q$.

5.2.2 SQ-Index for image deblurring with Poisson noise

Under Poisson noise, we extend the variational denoising model to deblurring and denoising in the same time. The model for deblurring with Poisson noise is:

$$E(u) = |\nabla u| + \beta \langle Au - f \log Au \rangle.$$

(5.15)

The solution $u$ is the minimizer of $E(u)$. Since the model is convex, we can apply primal-dual method to solve it. The equivalent form of model (5.15) is:

$$E(u, v, p) = \langle \nabla u, p \rangle + \beta \langle v - f \log v \rangle.$$

(5.16)

Then numerically, we can solve a max-min problem:

$$\min_{u, v} \max_{\{p, \|p\| \leq 1, q\}} \langle \nabla u, p \rangle + \beta \langle v - f \log v \rangle + \langle Au - v, q \rangle$$

$$- \frac{1}{2\tau} \|p - p^k\|^2 - \frac{1}{2\tau} \|q - q^k\|^2 + \frac{1}{2\tau} \|u - u^k\|^2 + \frac{1}{2\tau} \|v - v^k\|^2.$$  (5.17)

The solution of (5.17) is:

$$\tilde{p}^{k+1} = p^k + \tau \nabla u^k;$$

$$p^{k+1} = \frac{\tilde{p}^{k+1}}{\max(1, |\tilde{p}^{k+1}|)};$$

$$q^{k+1} = q^k + \tau (Au^k - v^k);$$

$$u^{k+1} = u^k + \tau (\text{div} p^{k+1} - A^T q).$$  (5.21)
\[ v^2 + (\beta \tau - v^k - \tau q)v - \tau \beta f = 0, \] (5.22)

where \( v \) is the solution of (5.22). To select parameter \( \beta \), we still iterate \( \beta \) the same way as in Poisson denoising. We calculate each SQ-Index for each \( \beta \) and choose the \( \beta \) which corresponds the biggest SQ-Index. The performance of Poisson denoising under Gaussian blur is in Figure 5.6 and Table 5.4. From the observation of the sensitivity of \( q \) for images, we set \( q = 1.15 \).

<table>
<thead>
<tr>
<th>image</th>
<th>PSNR</th>
<th>PSNR((q=1.15))</th>
<th>beta</th>
<th>beta((q=1.15))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>24.00</td>
<td>23.90</td>
<td>5.8</td>
<td>5.4</td>
</tr>
<tr>
<td>Parrot</td>
<td>23.97</td>
<td>23.97</td>
<td>5.8</td>
<td>5.8</td>
</tr>
<tr>
<td>Fluocells</td>
<td>29.18</td>
<td>29.18</td>
<td>5.8</td>
<td>5.8</td>
</tr>
<tr>
<td>Boat</td>
<td>26.36</td>
<td>26.36</td>
<td>5.8</td>
<td>5.8</td>
</tr>
</tbody>
</table>

From the performance of image deblurring under Poisson noise, we observe that the sensitivity of \( q \) is not very sensitive. The requirement of \( q \) is less strict than that for Poisson denoising case. It means that there is a large scope of \( q \) which can bring a good performance for deblurring under Poisson noise. Therefore, we conclude that SQ-Index is very efficient to select parameters for image deblurring under Poisson noise.
Figure 5.6: Column 1: image restored with best PSNR-value (try and trial) which is corrupted by Gaussian blur with variance 1 and Poisson noise with noise intensity 8; Column 2: image restored with model parameter selected by SQ-Index; Column 3: sensitivity of $q$ with respect to PSNR of restored image.
Table 5.5: Performance of Rician noise removal with parameter selected by SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>noise level</th>
<th>PSNR</th>
<th>PSNR sq</th>
<th>$\gamma$</th>
<th>$\gamma$ sq</th>
<th>RE of PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brain</td>
<td>20</td>
<td>27.58</td>
<td>27.30</td>
<td>33</td>
<td>25</td>
<td>1.02%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>24.87</td>
<td>24.87</td>
<td>49</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>PeMRI</td>
<td>20</td>
<td>28.79</td>
<td>28.79</td>
<td>34</td>
<td>33</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>26.18</td>
<td>26.18</td>
<td>50</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>Transverse</td>
<td>20</td>
<td>27.38</td>
<td>27.34</td>
<td>37</td>
<td>34</td>
<td>0.15%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>24.43</td>
<td>24.42</td>
<td>54</td>
<td>56</td>
<td>0.04%</td>
</tr>
</tbody>
</table>

5.3 SQ-Index for Rician noise removal model parameter selection

5.3.1 SQ-Index for image denoising with Rician noise

From Chapter 2, we introduce MAP-DCA model for Rician noise removal. Now we try to make image restoration by automatically choosing regularization parameter of the model. The model is:

$$\hat{u} = \arg \inf_u \gamma * \frac{1}{2\sigma^2} \int_\Omega u^2 dx - \gamma * \int_\Omega \log I_0(\frac{f u}{\sigma^2}) dx + \int_\Omega |\nabla u| dx$$  (5.23)

Model (5.23) is equivalent with the MAP model introduced in Chapter 2. To select parameter $\gamma$, we set $\gamma = 1 : 100$ and calculate SQ-Index with each $\gamma$. We select the $\gamma$ which brings the biggest SQ-Index. Now we show the performance of Rician noise removal with automatically selected parameter by SQ-Index. We set $q = 1.15$. The performance is in Figure 5.3.1 and Table 5.5. From the table and the figure, we see that the performance of image denoising under Rician noise with parameter selected by SQ-Index is close to the best possible PSNR. We can say that the SQ-Index is efficient to choose parameters for Rician denoising model.
5.3.2 SQ-Index for image deblurring with Rician noise

Now we show the performance of SQ-Index for choosing parameters in MAP model for image deblurring under Rician noise. We use the algorithm in Chapter 2 for image deblurring under Rician noise. We still compare the PSNR value with the best PSNR value which we iterate and choose by try and trial. Performance of image deblurring with SQ-Index is in Figure 5.8, 5.9 and Table 5.6. We set $q = 1$ for deblurring case for both Gaussian blur and motion blur.
Figure 5.8: Column 1: image corrupted with Rician noise with noise intensity 15 under Gaussian blur with variance 1; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index.
Figure 5.9: Column 1: image corrupted with Rician noise with noise intensity 15 under motion blur; Column 2: image restored with best PSNR by try and trial as reference; Column 3: image restored with parameter selected by SQ Index.

From Figure 5.8, 5.9 and Table 5.6, we can see that by SQ-Index, we can precisely choose the variational MAP model parameter $\gamma$ for restoring image with blur and Rician noise. The performance is very close to the best PSNR by try-and-trial. The
Table 5.6: Performance of Rician noise removal under blurry condition with parameter selected by SQ-Index

<table>
<thead>
<tr>
<th>image</th>
<th>BlurType</th>
<th>PSNR</th>
<th>PSNR sq</th>
<th>γ</th>
<th>γ sq</th>
<th>RE of PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>Gaussian</td>
<td>25.84</td>
<td>25.51</td>
<td>64</td>
<td>71</td>
<td>1.3%</td>
</tr>
<tr>
<td></td>
<td>motion</td>
<td>25.307</td>
<td>25.27</td>
<td>54</td>
<td>46</td>
<td>0.13%</td>
</tr>
<tr>
<td>PeMRI</td>
<td>Gaussian</td>
<td>28.68</td>
<td>28.67</td>
<td>54</td>
<td>50</td>
<td>0.3%</td>
</tr>
<tr>
<td></td>
<td>motion</td>
<td>27.76</td>
<td>27.58</td>
<td>50</td>
<td>35</td>
<td>0.6%</td>
</tr>
<tr>
<td>Coronal</td>
<td>Gaussian</td>
<td>27.11</td>
<td>27.08</td>
<td>71</td>
<td>80</td>
<td>0.1%</td>
</tr>
<tr>
<td></td>
<td>motion</td>
<td>25.58</td>
<td>25.48</td>
<td>62</td>
<td>64</td>
<td>0.4%</td>
</tr>
<tr>
<td>Transverse</td>
<td>Gaussian</td>
<td>26.01</td>
<td>26.01</td>
<td>87</td>
<td>86</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>motion</td>
<td>24.02</td>
<td>23.88</td>
<td>89</td>
<td>100</td>
<td>0.25%</td>
</tr>
</tbody>
</table>

relative error is very tiny with best PSNR value for most restored images.

5.4 Conclusion

In this chapter, we show various applications of SQ-Index for different variational model parameter selection. The experiments show that SQ-Index is efficient and robust for different models’ parameters selection by choosing a good $q$. For the moment, we choose $q$ by training some images and observe the sensitivity of $q$. In other words, we corrupted images the same way with the images we want to test. Like we want to recover the image corrupted with Gaussian blur, we train some images which are degraded by Gaussian blur. Then we compare the PSNR of restored images for the training images with different $q$. We choose a $q$ which can give the best performance on average for the training images. Then we use this $q$ to recover the real images. The weakness of this metric is that it needs many iterations which spends much time. The advantage of this metric comparing with other existed parameter selection methods is that SQ-Index has an explicit form which is easy to apply for
different models. From the various experiments, SQ-Index demonstrates a wide utility to recover image with different kinds of blur and noise.
Chapter 6

Summary

The objective of this thesis is to research and develop new numerical algorithms to convex and non-convex variational models for image denoising and image deblurring.

The main contribution of this thesis is:

- We propose a new primal-dual-projection algorithm to solve directly TV constrained model. The new algorithm can directly estimate Gaussian noise level.

- We apply DCA method for MAP non-convex model for Rician noise removal. The method is more efficient in both quality of restored image and computation time compared with other non-convex methods and is very competitive with convex models. For non convex models, the different initial conditions can bring to different local optimals. For our experiments, the different initial values can all bring a satisfying result. Therefore, in our experiment, we choose the observed image $f$ as our initial values which is meaningful physically in PDE diffusion theory.

- We apply DCA method for MAP non-convex model for Cauchy noise removal. The method performs competitively with the convex model while our model is less complicated and more applicable for real imaging problems.

- We define a new metric which can be used to select model parameters automatically. Our metric can be regarded as a no reference image quality assessment metric.

For our future work, we need to give more mathematical demonstration of SQ-Index and the selection of parameter scope for iteration is still by experience. We
need to further find an optimized algorithm to make SQ-Index compute faster. We can try to apply SQ-Index for more models’ parameter selection, like for removing impulse noise. For Poisson parameter selection, we can make more comparison like we can compare with discrepancy principle methods for Poisson noise removal proposed by Chan R [21]. For the DCA theory, we can further apply it for other noise removal, for instance, for speckle noise removal and even for other image processing subjects. We can also study the influence of different decomposition methods for $G(u)$ and $H(u)$ to make the algorithm more powerful and efficient.
References


diffusion tensor mri. In *International Conference on Medical Image Computing

[8] Matthew James Beal. *Variational Algorithms for Approximate Bayesian Infer-


[10] Mario Bertero, Patrizia Boccacci, Giorgio Talenti, Riccardo Zanella, and Luca
Zanni. A discrepancy principle for poisson data. *Inverse problems*, 26(10):105004,
2010.

global phase coherence. In *2012 IEEE International Conference on Acoustics,

phase coherence of an image. In *2008 15th IEEE International Conference on

[13] Peter Blomgren and Tony F Chan. Modular solvers for image restoration prob-
lems using the discrepancy principle. *Numerical linear algebra with applications*,

[14] Peter Blomgren and Tony F Chan. Modular solvers for image restoration prob-
lems using the discrepancy principle. *Numerical Linear Algebra with Applica-

530, 2005.


[73] Václav Smudl and Anthony Quinn. The variational bayes method in signal processing (signals and communication technology), 2005.


CURRICULUM VITAE

Academic qualifications of the thesis author, Mrs CUI Lei:

- Received the degree of Master of Finance, Economic and Management from Ecole des Ponts et Chaussés in France, 2011.

- Received the degree of Ecole Polytechnique (Master, Cycle Ingenieur) in France, 2011.

- Received the degree of Bachelor of Science in Electronic Engineering from Fudan University in China, 2007.

November 2016