

School of Electrical Engineering, Computing and Mathematical Sciences

**Image Processing by Variational Methods, Stochastic Programming and
Deep Learning Techniques**

**Lu Tan
0000-0002-3361-3060**

**This thesis is presented for the degree of
Doctor of Philosophy
of
Curtin University**

August 2020

Declaration

I hereby declare that this submission is my own work that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other or diploma of the university or other institutes of higher learning, except where due acknowledgment has been made in the text. I have obtained permission from the copyright owners, where necessary, to use any third-party copyright material reproduced in the thesis, or to use any of my own published work in which the copyright is held by another party.

Signature: _____

Date: _____

Abstract

Many outstanding image approaches in computer vision have been proposed and significant advances have been achieved. Variational methods and deep learning techniques are two excellent branches which have shown strong performance in many applications with great achievements. In addition, stochastic programming also shows great potential to improve existing computer vision techniques because of its solid mathematical foundations in probability and stochastic processes. In the area of variational methods, continuous multivariate models originally established for physical phenomena are commonly used for image processing tasks in recent decades, which have also resulted in solid mathematical theories. Stochastic programming provides useful tools to design and control stochastic systems using mathematical programming techniques, with the purpose to make optimal decisions under uncertainty. Large sets of applications have proved its value in applied science. In the field of deep learning, the trained neural network enables machines to make decisions comparable, or with higher accuracy than human in image classification and recognition. A huge amount of research was conducted on natural and medical images resulting in significant progress and promising outcomes. In recent years, convolution neural network (CNN) based deep learning approaches have attracted broad attention and gradually become dominant for solving a large amount of problems in various image tasks.

The aim of this thesis is to investigate some effective approaches to tackle different problems in computer vision. In particular, we first study the effectiveness of variational and PDE based methods for illusory contour reconstruction and image segmentation, and then design the corresponding optimization algorithms for efficiency improvement. Next, we formulate variational image segmentation problems in the optimization framework of stochastic programming, tackling diverse segmentation problems with random noises. In the third part, we focus on exploring the fusion approaches integrating variational models and deep neural networks for challenging image tasks using unsupervised and supervised learning respectively. It is expected that the integration of these two approaches is able to result in significant performance gains.

The first focus of this thesis is the investigation of variational and PDE based meth-

ods, we discuss capabilities of several constraints in important features preserving, image information utilizing and significant gains of introducing some famous mathematical theory. The following problems are addressed: (i) color image restoration and inpainting via multi-channel total curvature; (ii) Euler’s elastica based illusory contour reconstruction and image segmentation; (iii) nonlinear diffusion based image segmentation. Extensive experiments are conducted on different images to validate the proposed approaches as well as the merits of the designed algorithms in terms of computational efficiency.

The second focus of this thesis is to convert the monotone stochastic variational inequality problems in mathematics to diverse problems of unsupervised image segmentation with random noises by transforming the original minimization problems into the optimization framework of a two-stage stochastic programming. As far as I know, it is the first attempt to apply stochastic programming in image tasks. Then a general numerical algorithm based on progressive hedging algorithm (PHA) is proposed. Technically, all the sub-problems derived from the framework of PHA can be solved by using the curvature-weighted approach and the convex relaxation method. Then, an alternating optimization strategy is applied by using some powerful accelerating techniques. Extensive experiments have been conducted on both synthetic and real images, which displayed significant gains of the proposed segmentation models and demonstrated the advantages of the developed algorithms.

The third focus of this thesis lies in the investigation of deep variational architectures, integrating the good properties of both variational and deep learning techniques in order to produce results with higher performance. Two main approaches are proposed by introducing three different integration frameworks built on variational energy functional and CNN-oriented deep networks. One is an unsupervised model with applications in image restoration, inpainting, decomposition and texture segmentation. Another is designed to segment the prostate gland from a MRI dataset including different prostate images with limited resolution and quality. Technically, a 3D CNN model V-net is designed specifically for volumetric medical images is set for the first phase. Then the detected prostate-like map is then passed to the next phase for optimization via variational approach.

In summary, we propose three main approaches for several image processing problems motivated by variational methods, stochastic programming and deep learning techniques in this thesis. Some future work will lie in investigating robust approaches for challenging image understanding tasks.

Acknowledgments

This thesis is the completion of my PhD study, which also recorded all my efforts devoted for this long journey. Honestly it was a rather tough way for me with confusion and frustration at one time. But the unconditional encouragement and supporting from the people all around me gave me endless strength to move on and overcome difficulties. They made an indispensable contribution to accomplish this huge task. I would like to take this opportunity to thank all the following people for their great guidance and support throughout the years of my PhD program.

- First and foremost, I wish to express my deepest gratitude to my supervisor Professor Ling Li and co-supervisor Associate Professor Wanquan Liu at this moment of accomplishment, for taking me as a PhD student and for all your constant guidance, support and encouragement through my study period. This work would not have been possible without your involvement and great advice. You let me see more of the whole picture of research and look at problems more holistically. This valuable learning period has been a memorable experience that will be of endless benefit to my whole life.
- I would like to express a special thanks to Curtin University for providing financial support for my PhD study through Curtin International Postgraduate Research Scholarship (CIPRS) and Research Stipend Scholarship.
- I am greatly grateful to Prof. Jie Sun, Dr. Senjian An and Dr. Kylie Munyard, for taking the time to discuss with me, provide suggestions and help me understand what I needed to focus on. You brought my research work new ideas, professional experience and useful methods, I learned so much from each of you.
- I would like to appreciate and acknowledge the support, friendships and collaboration received from all my fellow research students and researchers (Antoni Liang, Min Zhang, Qilin Li, Guojia Hou, Huizhu Pan) for many discussions and inspirations. You have taught me many lessons about not only work but also life.

- I also greatly appreciate the support from my parents. Thanks for accompanying me grow up, teaching me a lot about life and believing in me all the time. I send a very special thank-you to my parents in law, for your love, understanding and moral support through the time of pursuing my PhD degree. I am so grateful to have you all in my life.
- I would also like to thank our administrative officer Mary Mulligan, Ether Yew, and Cindy Wong, for your assistance on the paper works.
- Lastly, I owe thanks to the most important person, my husband, Long Liu, for everything you did for me and always standing behind me as a pillar of strength. Thank you for every beautiful moment spent with me and every wonderful memory created with me, for you are here with me.

Publications

This is a list of works that have been published over the course of the author's PhD Degree in chronological order:

- **Tan, L.**, Pan, Z., Liu, W., Duan, J., Wei, W., & Wang, G. (2018). Image Segmentation with Depth Information via Simplified Variational Level Set Formulation. *Journal of Mathematical Imaging and Vision*, 60(1), 1-17.
- **Tan, L.**, Liu, W., & Pan, Z. (2018). Color Image Restoration and Inpainting via Multi-Channel Total Curvature. *Applied Mathematical Modelling*, 61, 280-299.
- Liu, X., Wei, W., & **Tan, L.** (2018). Realization of Detection and Location of Standing Face in Intelligent Recording and Broadcasting System Computer Engineering and Applications. *Journal of Qingdao University (Natural Science Edition)*, (1), 14.
- **Tan, L.**, Li, L., Liu, W., & Pan, Z. (2019). Fast Computational Approach for Illusory Contour Reconstruction, *Multimedia Tools and Applications*, 78(8), 10449-10472.
- **Tan, L.**, Liang, A., Li, L., Liu, W., Kang, H., & Chen, C. (2019). Automatic Prostate Segmentation Based on Fusion Between Deep Network and Variational Methods. *Journal of X-Ray Science and Technology*, 27(5), 821-837.
- **Tan, L.**, Li, L., An, S., & Pan, Z. (2019). Nonlinear diffusion based image segmentation using two fast algorithms. *Mathematical Foundations of Computing*, 2(2), 149-168.
- **Tan, L.**, Li, L., Liu, W., Sun, J., & Zhang, M. (2020). A Novel Eulers Elastica-Based Segmentation Approach for Noisy Images Using the Progressive Hedging Algorithm. *Journal of Mathematical Imaging and Vision*, 62(1), 98-119.
- **Tan, L.**, Li, L., Liu, W., An, S., & Munyard, K. (2020). Unsupervised Learning of Multi-task Deep Variational Model. (*under review by supervisors*)

- Pan, H., Song, J., Liu, W., Li, L., Zhou, G., **Tan, L.**, & Chen, S. (2020). Using the Split Bregman Algorithm to Solve the Self-Repelling Snake Model. *arXiv preprint arXiv:2003.12693*.
- Li, X., Hou, G., **Tan, L.**, Liu, W., & Huang, B. (2020). A hybrid framework for underwater image enhancement by dehazing. *Signal Processing: Image Communication*. (*under review*)
- Hou, G., Zhao, X., Pan, Z., Yang, H., **Tan, L.**, & Li, J. (2020). Benchmarking Underwater Image Enhancement and Restoration, and Beyond. *IEEE Access*, 8, 122078-122091.

Patents

This is a list of patents that have been issued over the course of the author's PhD Degree in chronological order:

- Wei, W., Liu, X., **Tan, L.**, Ji, Y., & Wang, J. (issued December 15, 2017, granted April 7, 2020). An Approach for Face Detection and Location from a Standing Position in the Intelligent Recording and Broadcasting System. CN Patent 201710632091.4. (in Chinese)

Attribution Statement

Chapters 3 to 6 of this thesis are based on works that have been published with joint-authorship. We hereby make an authorship attribution statement to clarify the contribution of individual authors.

Chapter 3 is based on the publications:

- **Tan, L.**, Liu, W., & Pan, Z. (2018). Color Image Restoration and Inpainting via Multi-Channel Total Curvature. *Applied Mathematical Modelling*, 61, 280-299.

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	60%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Wanquan Liu)	×	✓	×	×	✓	✓	20%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Zhenkuan Pan)	✓	×	×	×	✓	✓	20%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

- **Tan, L.**, Li, L., Liu, W., & Pan, Z. (2019). Fast Computational Approach for Illusory Contour Reconstruction, *Multimedia Tools and Applications*, 78(8), 10449-10472.

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	70%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Ling Li)	×	×	×	×	✓	✓	10%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Wanquan Liu)	×	×	×	×	✓	✓	10%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 4 (Zhenkuan Pan)	✓	×	×	×	✓	×	10%
Co-author 4 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

- **Tan, L.**, Li, L. An, S., & Pan, Z. (2019). Nonlinear diffusion based image segmentation using two fast algorithms. *Mathematical Foundations of Computing*, 2(2), 149-168.

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	60%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Ling Li)	×	×	×	×	✓	✓	10%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Senjian An)	×	×	×	×	✓	✓	10%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 4 (Zhenkuan Pan)	✓	×	×	×	✓	✓	20%
Co-author 4 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

Chapter 4 is based on the publication:

- **Tan, L.**, Li, L., Liu, W., Sun, J., & Zhang, M. (2020). A Novel Eulers Elastica-Based Segmentation Approach for Noisy Images Using the Progressive Hedging Algorithm. *Journal of Mathematical Imaging and Vision*, 62(1), 98-119.

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	60%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Ling Li)	✓	×	×	×	✓	×	10%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Wanquan Liu)	✓	×	×	×	×	✓	10%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 4 (Jie Sun)	×	×	×	×	✓	✓	10%
Co-author 4 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 5 (Min Zhang)	×	×	×	×	✓	✓	10%
Co-author 5 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

Chapter 5 is based on the publication:

- **Tan, L.**, Li, L., Liu, W., An, S., & Munyard, K. (2020). Unsupervised Learning of Multi-task Deep Variational Model. (*under review by supervisors*)

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	60%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Ling Li)	×	×	×	×	✓	✓	10%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Wanquan Liu)	×	×	×	×	✓	✓	10%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 4 (Senjian An)	✓	×	×	×	✓	×	10%
Co-author 4 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 5 (Kylie Munyard)	×	✓	×	×	✓	×	10%
Co-author 5 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

Chapter 6 is based on the publication:

- **Tan, L.**, Liang, A., Li, L., Liu, W., Kang, H., & Chen, C. (2019). Automatic Prostate Segmentation Based on Fusion Between Deep Network and Variational Methods. *Journal of X-Ray Science and Technology*, 27(5), 821-837.

	Conception and Design	Data Acquisition and Manipulation	Programming	Experiments	Interpretation and Discussion	Manuscript Writing and Revision	Total Contribution
Co-author 1 (Lu Tan)	✓	✓	✓	✓	✓	✓	35%
Co-author 1 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 2 (Antoni Liang)	✓	✓	✓	✓	✓	✓	35%
Co-author 2 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 3 (Ling Li)	✓	×	×	×	✓	✓	10%
Co-author 3 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 4 (Wanquan Liu)	✓	×	×	×	✓	✓	10%
Co-author 4 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 5 (Hanwen Kang)	×	×	×	×	✓	×	5%
Co-author 5 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							
Co-author 6 (Chao Chen)	×	×	×	×	✓	×	5%
Co-author 6 Acknowledgement: I acknowledge that these represent my contribution to the above research output Signed:							

Contents

1	Introduction	1
1.1	Research Objectives	4
1.2	Thesis Structure and Contribution	6
2	Background	9
2.1	Variational Techniques	9
2.1.1	Partial Differential Equations	9
2.1.2	Level Sets	11
2.1.3	Mumford-Shah Model	14
2.1.4	Euler’s Elastica Constraint	16
2.1.5	Variational Image Processing	19
2.2	Stochastic Programming Theory	25
2.2.1	Two-stage Stochastic Programming	26
2.2.2	The Progressive Hedging Algorithm for Two-stage Stochastic Programming	27
2.2.3	Convergence properties of the PHA	30
2.3	Deep Learning models	30
2.3.1	Deep Feedforward Network	31
2.3.2	Convolutional Neural Networks	32
2.4	Summary	37
3	Variational and PDE Based Methods for Diverse Image Tasks	38
3.1	Introduction	38
3.2	Proposed Multi-Channel Total Curvature Model for Image Restoration	45
3.2.1	The Functional of Variational MTC- ℓ_1 Model and its GDM method	45

3.2.2	Application of ADMM to MTC- ℓ_1 Model	46
3.3	Proposed Simplified Variational Level Set Formulation for Illusory Contour Reconstruction	56
3.3.1	A New Formulation for Illusory Contour Capture and its GDM method	56
3.3.2	ADMM-P Algorithm for Illusory Contour Capture model	58
3.4	Proposed Nonlinear diffusion based models for Image Segmentation	61
3.4.1	Nonlinear Diffusion based Image Segmentation	61
3.4.2	Design of ADMM and NVPM for the Nonlinear Diffusion Based Image Segmentation.	62
3.5	Experiments	67
3.5.1	MTC- ℓ_1 model for color image processing	67
3.5.2	Illusory Contour Capture	76
3.5.3	Nonlinear Diffusion based Image Segmentation	83
3.6	Summary	88
4	Stochastic Programming Based Variational Segmentation Approach for Noisy Images	90
4.1	Introduction	90
4.2	Novel Formulations for Different Segmentation Problems Incorporating Stochas- tic Noises Via Progressive Hedging	93
4.2.1	Two-phase Segmentation Based Application	95
4.2.2	PHA with ADMM-C Algorithm for Two-phase Segmentation Appli- cation	96
4.2.3	Segmentation with Depth Based Application	102
4.2.4	PHA with ADMM-C Algorithm for Segmentation with Depth	103
4.2.5	Termination Criteria	108
4.3	Experiments	110
4.3.1	Experiments for Two-Phase Segmentation on Synthetic Images	110
4.3.2	Experiments for Two-Phase Segmentation on Real Images	113
4.3.3	Experiments for Segmentation with Depth on Synthetic Images	114
4.3.4	Experiments for Segmentation with Depth on Real Images	117
4.4	Summary	119

5	Multi-task Deep Variational Model for Denoising, Inpainting, Decomposition and Segmentation	121
5.1	Introduction	121
5.2	Related Work	124
5.2.1	Variational Methods	124
5.2.2	Convolutional Neural Network (CNN) based Approaches	128
5.3	Proposed Multi-task Model	131
5.3.1	Loss Function Inspired by Multi-Task Variational Functional	131
5.3.2	Multi-Task Deep Variational Architectures	133
5.4	Experimental Results	138
5.4.1	Experiments for Denoising	138
5.4.2	Experiments for Inpainting	142
5.4.3	Experiments for Decomposition	144
5.4.4	Experiments for Simultaneous Implementation of Denoising, Inpainting and Decomposition	146
5.4.5	Experiments for Simultaneous Implementation of Image Segmentation with Effects of Multiscale, Structure-Texture Separation and Restoration	147
5.4.6	Comparison with double-DIP networks	149
5.5	Summary	151
6	Deep Variational Learning for Prostate Segmentation	152
6.1	Introduction	152
6.2	Related Work	155
6.2.1	Learning-based Approaches	155
6.2.2	Variational Methods	156
6.3	The Proposed Approach	157
6.3.1	V-Net and Training Configuration	158
6.3.2	Variational Approach for Prostate Segmentation	161
6.3.3	Platform	164
6.3.4	Numerical Evaluation	164
6.4	Experiments	166

6.4.1	Experiments for Comparison with Pure Variational Methods	167
6.4.2	Experiments for Comparison with the Original V-Net	168
6.4.3	Experiments for Comparison with V-Net Integrated with Global His- togram Matching	169
6.5	Summary	170
7	Conclusion and Future works	172
7.1	Conclusion	172
7.2	Future Works	175

List of Figures

1-1	Examples for various types of challenging problems in reconstructing and recognizing integrated shapes in computer vision.	2
2-1	Spatial instruction for pixel location.	11
2-2	Graph explanation for level sets.	12
2-3	Regularization of the Heaviside function and Dirac function.	14
2-4	Euler’s elastica used for image denoising.	17
2-5	Euler’s elastica used for image segmentation.	18
2-6	Euler’s elastica used for segmentation with depth.	19
2-7	The list of application areas of stochastic programming.	26
2-8	A diagram for two-stage stochastic programming.	28
2-9	An example of one neuron.	31
2-10	An example of multi-layer perceptron (MLP).	33
2-11	An example of convolution operation.	34
2-12	An example of basic CNN architecture.	36
2-13	An example of CNN architecture with deconvolution.	37
3-1	The denoising effects of MTV- ℓ_1 model and our proposed MTC- ℓ_1 model.	69
3-2	The plots of parametric errors and the energy curve. (a-d) are obtained by our MTC- ℓ_1 model from Fig. 3-1(a3). (e-h) are obtained by our MTC- ℓ_1 model from Fig. 3-1(b3).	71
3-3	Comparisons of energy curves. (a): energy curves of using GDM and ADMM for example shown in Fig. 3-1(a3); (b): energy curves of using the GDM and ADMM from Fig. 3-1(b3).	72
3-4	The inpainting effects of MTV- ℓ_1 model and our proposed MTC- ℓ_1 model.	73

3-5	The plots of parametric errors and the energy curve. These plots are obtained by our MTC- ℓ_1 model for image inpainting from Fig. 3-4(a3).	74
3-6	Comparisons of energy curves. (e): energy curves of using the GDM and ADMM for example shown in Fig. 3-4(a3).	75
3-7	Scale spaces generated by MTC- ℓ_2 model and our MTC- ℓ_1 model.	76
3-8	The denoising effects of LTC- ℓ_1 model and our MTC- ℓ_1 model.	78
3-9	The procedure for illusory contour capture.	79
3-10	Illusory contour capture for synthetic images.	80
3-11	Illusory contour capture for noisy images.	81
3-12	Illusory contours capture for three real images.	82
3-13	Illusory contours capture for large real images.	83
3-14	Effects of our model.	85
3-15	Effects of GAC and PSAC model.	86
3-16	Effects of our model, GAC model and PSAC model.	87
3-17	Non-threshold solutions of our methods. The first column: results of ϕ . The second column: zoomed small sub-regions (red rectangles in (c1) and (d1)) for comparison.	87
3-18	Effects of our model, GAC model and PSAC model on colour images.	88
3-19	Plots of parametric errors and energy curves. The first row is obtained by our model using ADMM. The second row is obtained by our model using NVPM*.	88
4-1	Example of PHA in stochastic programming for segmentation improvement. (a): original image; (b): noisy image; (c): results obtained by CV model; (d): results obtained by CV model with PHA in stochastic programming.	94
4-2	Two phase segmentation for synthetic noisy images with incomplete shapes and letters. (a) and (e): original images, noisy images and the same initial ϕ^0 ; (b) and (f): results obtained by CV model; (c) and (g): results obtained by CVE model; (d) and (h): final results from our proposed model (4.2).	112
4-3	Segmentation results from our model (4.2) without the last two terms ($v(\cdot)$ term and r -term).	113

4-4	The relative residual plots according to the equations in Section 4.3.5. (a): auxiliary variables in R_{τ}^k ; (b): auxiliary variables in $R_{\bar{w}_{\xi}}^k$; (c): Lagrange multipliers in $R_{v_{\xi}}^k$; (d): Lagrange multipliers in $R_{\lambda_{\xi}}^k$; (e): optimal solution $R_{\phi_{\xi}}^k$; (f): the energy functional.	113
4-5	Two phase segmentation for real noisy images with incomplete shapes. (a) and (g): noisy images; (b) and (h): initial ϕ^0 ; (c) and (i): results obtained by CV model; (d) and (j): results obtained by CVE model; (e) and (k): final results from our proposed model (4.3); (f) and (l): intermediate curve evolution by our proposed model.	115
4-6	Traditional segmentation with depth for an image with two circles. (a): noisy images and results obtained by the standard multiphase segmentation model; (b): the initialization for two binary functions ϕ_h^0 ; (c): results obtained by traditional segmentation with depth model [13,15].	117
4-7	Our proposed model (4.19) for an image with two circles. (a): noisy images and results by the standard multiphase segmentation in stochastic programming model; (b): the initialization for two binary functions ϕ_h^0 ; (c): results obtained by our proposed model	117
4-8	Our proposed model (4.20) for real image. (a) and (d): noisy images and results by the standard multiphase segmentation in stochastic programming model; (b) and (e): the initialization for two binary functions ϕ_h^0 ; (c) and (f): results by our proposed model.	118
4-9	Histograms of the binary level set functions (non-threshold and threshold).	119
5-1	Basic architecture of DIP.	129
5-2	Reduced version of the framework without decomposition capability. For TGV, the module in dashed box need to be included into ℓ_1 -based $\zeta(u)$. For other models, it should be omitted. . . .	133
5-3	Full version of the framework. For TGV-type, the module in dashed box need to be included into ℓ_1 -based $\zeta(u)$. For other models, it should be omitted.	134
5-4	Texture Segmentation with Multiscale and Elastica Constraints Framework.	136
5-5	Comparison of the effects between LTC [165], MTC [165], DIP [170] and our proposed model. The detail differences are highlighted via rectangles. LTC and MTC results were presented in [165], which are put here directly. . . .	141

5-6	Comparison of the effects between MTV [189], MTC [165], DIP [170] and our proposed model (one reduced version and one full version). Some differences are highlighted via rectangles. MTV and MTC results used here were taken from [165].	142
5-7	Comparison of the effects between [206] and our proposed RV-EE model. Target Images and Results used here were initially presented in [206]. . . .	143
5-8	Comparison of the image decomposition between EWSO model [54] with various parameter settings (under each image) and our proposed model. (a)-(e) are taken from [54]	144
5-9	Zoomed small sub-regions of the results from EWSO model taken from [54] with the best parameter setting and our proposed model for detail comparison. The detailed differences are highlighted via rectangles in three colors. .	145
5-10	Comparison of the effects between Kim et al.'s work [86] and our proposed model (FV-MTL, FV-MBH and FV-MTC) for decomposition in colorscale. (a), (d) and (g) are taken from [45] and (b), (e) and (h) are the results from [86]	146
5-11	Our proposed model (full version) for simultaneous implementation of denoising, inpainting and decomposition in color.	147
5-12	Performance comparison between [202] and our proposed model (5.19) for multiscale and object segmentation with textures, impurities or noises. Original images (a)-(b) and Results (c)-(e), (i) presented here were directly taken from [202].	149
5-13	Performance comparison between Double-DIP [58] and our proposed model (FV-TL, FV-BH, FV-TGV, FV-EE and FV-TC) for texture segmentation.	150
6-1	The flowchart illustrates how to implement our approach.	158
6-2	Global histogram matching.	160
6-3	Various examples of prostate MRI for testing.	161
6-4	3-D reconstructions on: (a) Original V-Net. (b) V-Net with global histogram matching. (c) Pure variational models. (d) Our proposed framework. (e) Ground truth.	167
6-5	The effects of variational models and our proposed approach from patient 6. (a): Original MRI. (b): Variational results. (c): Our results. (d)-(e): Details.	168

6-6	Comparison of patient 46. (a): Original MRI. (b): Ground truth. (c): Original V-Net results. (d): V-Net integrated with global histogram matching results. (e): Our proposed complete framework results.	169
6-7	The effects of pure improved V-Net and our proposed approach from patient 29. (a): Original MRI. (b): Results from V-Net integrated with global histogram matching (blue line), our final results (yellow line) and the ground truth (red line).	170

List of Tables

2.1	Potential functions for the regularization term.	24
2.2	Potential functions of different noise distributions.	25
2.3	Formulations and plots of different active functions.	32
3.1	Comparisons of PSNR, iterations and computational time.	69
3.2	Comparisons of PSNR, iterations and computational time.	75
3.3	Comparisons of PSNR, iterations and computational time.	76
3.4	Number of iterations and time costs using GDM and ADMM-P methods. .	78
3.5	Number of iterations and time costs using GDM and ADMM-P methods. .	79
3.6	Number of iterations and time costs using GDM and ADMM-P methods. .	81
3.7	Number of iterations and time costs using GDM and ADMM-P methods. .	82
3.8	Comparisons of iterations and time using different methods.	85
4.1	Potential functions of different noise distributions for segmentation with depth application.	106
4.2	Minimal energies of different ordering relations.	117
4.3	Minimal energies of different ordering relations for Fig. 4-8(c).	118
4.4	Minimal energies of different ordering relations for Fig. 4-8(f).	119
4.5	Number of iterations and computational time.	119
5.1	High-order Energy Functionals	126
5.2	Average SNR for denoising comparison (grayscale).	139
5.3	Different regularization terms in frameworks of LTV and MTV	140
5.4	Comparison of PSNR for Pepper and Salt noise removal (color synthetic), and the difference between coupling methods using layered-channel and multi- channel	141

6.1	Chosen Evaluation Metrics.	165
6.2	Seven evaluation metrics on (a) Variational Method, (b) V-Net, (c) V-Net+Histogram matching, (d) The proposed Method. Numbers in bracket represent the standard deviation.	166

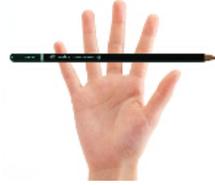
Chapter 1

Introduction

Occlusion, texture analysis, intensity homogeneity and illusory contours of target objects are some of the challenging problems in reconstructing and recognizing integrated shapes correctly in image understanding, some examples are shown in Fig.1-1. Although much research effort has been invested on these problems, current approaches based purely on low-level information such as intensity, colour and textures show limited ability on tackling them, especially when images come in extremely low quality, e.g., with large amount of noise or damage and limited resolution.

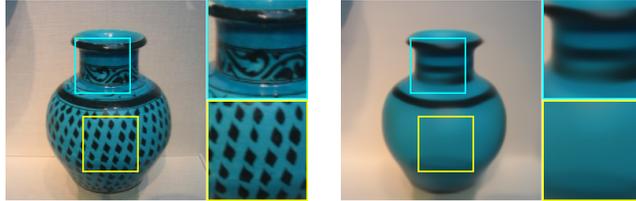
A desirable property for image processing algorithms is to restore images without losing important features such as edge sharpness, smoothness, corner and contrast etc. Such pre-processing is a crucial step in image analysis. In the last two decades, there have been many variational models [8, 39, 126, 148] proposed to achieve such a goal. The classical first order Total Variation (TV) model or Rudin-Osher-Fatemi (ROF) model [140] for image denoising can be considered as the foundation of such models due to their edge preserving capability. For example, image inpainting [15] and decomposition models [172] were proposed based on the Total Variation model. However there are some weakness in the TV model, such as the staircase effects and the smearing of corners. In order to ameliorate the staircase effect, various higher order variation models have been proposed [33, 104, 167]. The curvature-based restoration models [158, 209] are able to preserve edges and corners while eliminating the staircase effect. Among all the curvature-based models, Eulers elastica model [160, 166, 212] not only has the capabilities mentioned above, but is also able to restore the missing part of edges or shapes in images, which is of great significance for image understanding with

Occlusion:



(a) Fingers are occluded

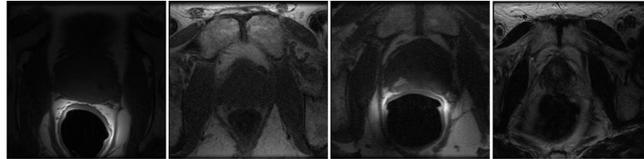
***Structure Extraction
from Texture:***



(b) Original image

(c) Structure

Intensity Homogeneity:



(d) Tissues and background share similar feature

Illusory Contours:



(e) Gaps of incomplete letters can be interpolated

Figure 1-1: Examples for various types of challenging problems in reconstructing and recognizing integrated shapes in computer vision.

problems such as occlusion, large damages or noises, etc.

Partial differential equations (PDE) based variational image processing originated in the 1980s and was well developed in the next few years. It works very well in preserving important image features, dealing with situations such as large intensity inhomogeneity, object occlusion etc, and is memory efficient in computation. Discrete image data as well as the associated operators are formulated in the perspective of continuation, whose solution is iteratively obtained by minimization of certain energy functional. A wide range of reliable numerical schemes for variational models are available, and they can be easily implemented and accelerated.

Variational methods were developed from mathematical analysis, which focused on investigating the extreme value of energy functional with respect to some unknown function.

A functional refers to a mapping relationship between a function set and a real number set and can be viewed as the function of functions. A variational problem is to search the extreme value of some specific functional. The solution of the functional extreme value problem is usually determined by minimizing a certain energy functional. When a basic variational model has been successfully established, some modifications or extensions can be made on this model to obtain results with improved performance and achieve wider applications. In general, the implementation of such modifications and extensions is straightforward and simple. However, there still exists limitations, e.g. the results are quite sensitive to numerous predefined parameters and heavily depend on initial settings and the number of iterations.

Unfortunately, uncertain information contained in images could be a tough problem for variation methods, which should be taken into full consideration. For example, existing variational segmentation models would fail to produce desired results on noisy images especially when the type of noise is unknown or there are more than one type of noise in the image. This is due to the fact that they all rely on a one-to-one mapping relationship between the parameters to be evaluated and the images with specific probability density distribution, without the capability to properly detect and deal with the random noises in the images. The aim of stochastic programming [16, 130, 150, 175] is to determine optimal responses to information involving uncertainty. With the rapid development of stochastic programming theory, a variety of models are formulated to solve the optimization problems with random data. Stochastic programming are useful in many practical situations. It would be a good way to tackle the tough problem caused by uncertainty by introducing stochastic programming theory into variational framework. Besides, general-purpose optimization approaches may not be able to apply directly. Designing applicable approaches to solve the associated problems is another important research direction.

In recent years, machine learning technology has been on its rise and drawn a lot of attention in computer science. Current studies on machine learning can deal with datasets of prior shapes with random geometries, and can cope with image noise and clutter, as well as partial occlusions. According to [92], conventional machine-learning techniques lack the ability of processing natural data in their raw form. Representation learning is one of the useful tools tackling this kind of limitation, which enables a machine to be fed with raw data and to automatically obtain the representations needed for different tasks. In the field

of deep learning, shared lower representation layers are defined and composed, then very complex features can be learned with excellent results, especially in image classification [2] and recognition [87]. A huge amount of research was conducted on natural images resulting in significant progress and promising outcomes. Among them, convolutional neural network (CNN) [52, 88, 152] is one of the most well-known approaches with significant achievements for a wide range of applications in computer vision and machine learning. CNN architecture especially works well on 2D image-based training datasets. Most of the state-of-the-art outcomes in computer vision usually rely on large amounts of available data. However, it is difficult for deep learning techniques to make good enough decisions when dealing with unseen data [14]. In addition, models based on deep learning are restricted to pixel-level processing. While variational methods perform very well in tackling these limitations. Deep learning techniques based on the convolutional neural network (CNN) and variational methods based on partial differential equations (PDE) are two excellent branches which have shown strong performance in many applications. It is expected that the integration of these two approaches is able to result in innovative ideas with performance gains.

In this thesis, we focus on investigating effective approaches motivated by classic variational methods, stochastic programming and deep learning techniques for different problems in computer vision. The effectiveness not only refers to higher accuracy and precision but also higher computational efficiency. One key observation in research is that the fusion approaches can bring in new thinking, new tools and developments. Variational methods and stochastic programming can be incorporated since they are both related to optimal decision making. Introduction of stochastic programming endows variational methods the additional ability to deal with uncertain data. The possibility of the fusion between variational methods and deep learning due to their similar process for obtaining optimal solution, which is to minimize the objective function (i.e., the energy functional in variational methods). The minimization of the energy functional can be done in a data-driven way using deep neural networks.

1.1 Research Objectives

The main purpose of this thesis is to develop different robust approaches to tackle the pixel-level image problems, including image processing, illusory contour capture, decomposition

and segmentation. After conducting an in-depth literature review in this direction, we identify several focuses of our research.

- **The first focus of this research is variational and PDE based methods for various image tasks.** In computer vision, there are some major challenges need to be tackled: Large noises or damages could easily influence the final results since they may hide important features; Occlusion and intensity homogeneity could result in serious difficulties. Illusory contours is a common problem caused which is difficult to be dealt with in standard segmentation approaches; Computational efficiency need to be improved. Therefore, the research objectives of this part are identified as follows.
 - (i) Image restoration such as denoising for removing noise, inpainting for restoring large damages: the proposed models will also be extended into colour images.
 - (ii) Illusory contour capture and illusory shape reconstruction: new models to be proposed, which not only aims to reconstruct the missing contours or shapes of the objects but also produce results with higher accuracy and more significant details.
 - (iii) Exploration of reliable mathematical techniques.

- **The second focus is to solve the above mentioned problems containing uncertainty by embedding stochastic property into variational energy functional.** Current variational segmentation models with capability of tackling noises are designed for specific noises. When the type of noise is unknown or there are more than one type of noise in the image, it would turn into a difficult problem. The research objectives of this part are given as follows.
 - (i) Solve problems of two-phase segmentation and segmentation with depth on images with unknown noises or arbitrary damages. Develop techniques with better properties in dealing with image feature information for further performance improvement.
 - (ii) Explore methods for solving such an optimization problem and the associated problems caused by the constraints derived from (i).

- **The third focus is to explore the advantages of deep variational learning approaches.** This part bridges the gap between two popular yet disconnected areas

in computer vision, the classic variational methods and deep learning. The approaches proposed here are to answer the questions:

- (i) How to incorporate the variational methods and deep learning to inherit the merits of both techniques?
- (ii) As a well-known approach in image processing research, can some useful features from variational methods still be beneficial for deep learning?
- (iii) For unsupervised learning, is it possible for deep learning architecture to provide prior information into the variational framework and greatly improve the performance? Can deep networks serve as the prior descriptors containing adequate image statistics?
- (iv) For supervised learning, can deep networks learn an initialization probability map for the variational methods? Can the learned probability reference give a helpful guidance for variational methods?

1.2 Thesis Structure and Contribution

In this thesis, the content of each chapter is briefly described as follows:

- **Chapter 2** introduces the background in three key aspects including the preliminary knowledge related to this thesis. In this chapter, we begin with the introduction of variational techniques including partial differential equations (PDE), the major effects of Eulers elastica constraint, the evolution of level set methods and the original Mumford-Shah model for image segmentation. Then some basic ideas of stochastic programming in the field of mathematical optimization are introduced briefly. Lastly the basics of deep learning are introduced, firstly the multi layer neural network (i.e. deep feedforward network), subsequently a short survey in terms of some recent advances in CNNs such as its basic architecture and applications. Developments of CNNs based on supervised learning and unsupervised learning are also provided.
- **Chapter 3** focuses on the three different types of variational energy functionals proposed for various image tasks and the fast numerical schemes optimizing each variational energy functional with high efficiency. **The contribution in this chapter:** (i) Image processing: a multi-channel total curvature model based on L1 norm (MTC-L1)

for vector-valued image restoration and inpainting is proposed. The alternating direction method of multipliers (ADMM) is introduced in this context; (ii) Illusory contour reconstruction: an equivalently reduced variational level set formulation is proposed by taking the level set functions as signed distance functions. An ADMM developed version is designed for this model; (iii) Image segmentation: a variational model is proposed for image segmentation based on active contours, nonlinear diffusion and level sets. Two fast numerical schemes are presented: ADMM comes first, followed by the normal vector projection method (NVPM). Experiments results validate the effectiveness of the proposed models and demonstrate their advantages by comparing with traditional methods.

- **Chapter 4** describes the incorporation of the classic variational methods and stochastic programming. **The contribution in this chapter** is that a novel variational formulation established by stochastic programming is proposed, which is able to properly deal with the random noises and improve the segmentation performance for noisy images. The Eulers elastica constraint is also embedded into the formulation to further enhance the segmentation performance. The proposed formulation of stochastic optimization is solved via the progressive hedging algorithm (PHA). Technically, all the sub-problems derived from the framework of PHA can be solved by using the curvature-weighted approach and the convex relaxation method. Another ADMM developed version is then applied by using some powerful accelerating techniques. Extensive experiments conducted on both synthetic and real-world images, and compared with existing classic models, demonstrate significant advantages of the proposed variational models.
- **Chapter 5** proposes the incorporation of the classic variational methods with deep learning techniques. **The contribution in this chapter** is that an unsupervised learning based multi-task deep variational model is proposed. It incorporates the CNN based deep image prior (DIP) architecture into the classic variational image processing models, which is presented in a unified form. The minimization problem solving strategy is transformed from iteratively minimizing the sub-problem for each variable to automatically minimizing the loss function by learning the generator network parameters. The proposed deep variational (DV) model contributes to the high-

order version and applications such as image restoration, inpainting, decomposition and texture segmentation, especially the simultaneous implementation of these tasks. The experimental results have demonstrated significant advantages of the proposed deep variational model.

- **Chapter 6** proposes a fusion approach between variational methods and deep learning techniques for medical image segmentation. Different from the DV model described in Chapter 5, the focus of this chapter is to establish a prostate MRI segmentation framework built on the well-known CNN architecture V-net, then a designed variational system is embedded into the architecture. Currently, the best results are obtained by CNN. However there still exist some problems when we apply CNN for the prostate segmentation, such as data distribution issue caused by insubstantial and inconsistent intensity levels and vague boundaries of the prostate in MRI. **The contribution in this chapter:** for segmenting the prostate gland from a MRI dataset including different prostate images with limited resolution and quality, we propose a prostate MRI segmentation framework based on the well-known CNN architecture V-net and the variational method in this Chapter. A global histogram matching approach is proposed to make intensity distribution of the MRI dataset closer to uniformity. In order to capture the real boundaries and improve the accuracy of segmentation, a module of variational models is employed to help improve the performance. Extensive experiments are conducted on a data set including different prostate images with limited resolution and quality, and the results validate the proposed approach, with better performance than the existing methods based on the original CNN or pure variational models.
- **Chapter 7** summarizes this thesis and discusses some potential directions for the future work.

Chapter 2

Background

This chapter starts with a basic description of the most common variational techniques, including partial differential equations (PDEs), level sets, Mumford-Shah model, Euler's elastica constraint and some popular image processing methods. Then a brief review of stochastic programming is presented, as well as the most widely studied and applied stochastic programming models, the two-stage programs. Lastly we introduce the deep learning models which are popularly used for computer vision problems, the deep feedforward network and convolutional neural network (CNN).

2.1 Variational Techniques

2.1.1 Partial Differential Equations

There is a long history for digital image processing with plenty of useful approaches proposed. It is hence difficult to say which approach is the most original. 1-D signal processing techniques might lay the foundation, utilizing filter theory, spectral analysis, or basic concepts of probability and statistics. In modern image processing, PDEs [8, 39, 126, 148] have become one of the most popular techniques with successful applications. As a famous tool which was intensively developed in image analysis since the 1990s, PDEs provide reliable calculus to solve variational problems. In addition, PDEs are also seen as a powerful tool to describe phenomena such as diffusion, transport, reaction, and so on. It can be usually observed that a PDE is established closely related to a specific physical problem. Researchers also found out that the physical background of PDEs performs well in image processing.

The basic concept of PDE based methods for image processing is that the image changes according to a certain PDE for the model defined in a continuous setting. Once the existence and the uniqueness of the solution for the PDE have been proven, the desired result can be obtained. Thus the primary task is to establish a reasonable PDE, that is to build a model. The common ways of such modelling are: (1) formulate the energy functional, then obtain its Euler-Lagrange equation (i.e. PDE) by variational methods; (2) draw an analogy between the expected changes on the image and some physical processes, for example, image smoothing and impurity diffusion, then build the corresponding PDE. After the model has been built, the most important step turns into how to solve the obtained PDE. However, the intrinsic discontinuity contained in the image associated functions, the nonlinearity of PDEs, and the huge amounts of image data may cause difficulties in numerical solution.

Finite-Difference Schemes

The ideal way for solving a PDE problem is that the analytical solution can be determined, since we can know precisely about all the details of the solution. Unfortunately such an ideal case is rare. Most of the time differential equations are very difficult to solve analytically or models are needed for computer simulations. It is necessary to obtain the approximation solution using some numerical schemes such as the finite difference, finite element method and spectral method. In image processing applications, the most common way to solve PDE is the finite difference, since images are in discrete digital format captured through uniform sampling. They naturally compose the uniform grid needed for the finite difference scheme. One essential point here is to implement discretization by transforming a continuous equation into an algebraic equation. Accordingly, the computational domain will be discretized into a grid which consists of multiple elements.

The central idea of the finite difference scheme is to approximate the partial derivative of a function with respect to some variable by the ratio of the difference in function values at two neighbour points to the distance between these two points. For instance, there are three difference expressions for the first-order partial derivative of function u at one point (i, j) in space:

$$\text{Forward difference: } \begin{cases} \partial_x^+ u_{i,j} = \left(\frac{\partial u}{\partial x}\right)^+ \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \\ \partial_y^+ u_{i,j} = \left(\frac{\partial u}{\partial y}\right)^+ \approx \frac{u_{i,j+1} - u_{i,j}}{\Delta y} \end{cases},$$

$$\text{Backward difference: } \begin{cases} \partial_x^- u_{i,j} = \left(\frac{\partial u}{\partial x}\right)^- \approx \frac{u_{i,j} - u_{i-1,j}}{\Delta x} \\ \partial_y^- u_{i,j} = \left(\frac{\partial u}{\partial y}\right)^- \approx \frac{u_{i,j} - u_{i,j-1}}{\Delta y} \end{cases},$$

$$\text{Central difference: } \begin{cases} \partial_x^0 u_{i,j} = \left(\frac{\partial u}{\partial x}\right)^0 \approx \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \\ \partial_y^0 u_{i,j} = \left(\frac{\partial u}{\partial y}\right)^0 \approx \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} \end{cases},$$

where $\Delta x = \Delta y = h$ is the uniform interval, and x, y denote coordinate directions. A spatial instruction is shown in Fig. 2-1 to illustrate the pixel location in the spatial digital image domain.

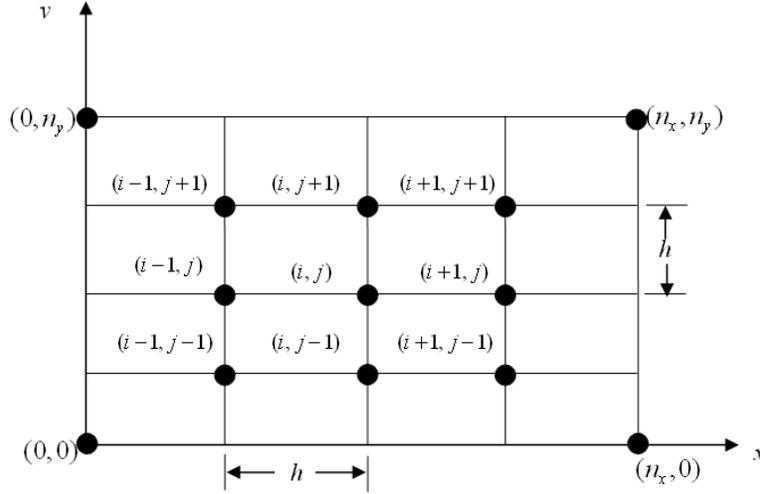


Figure 2-1: Spatial instruction for pixel location.

2.1.2 Level Sets

When conducting image segmentation using PDE, the problem is first modeled as an energy functional according to the processing requirements. Traditional way of minimizing the energy functional is to solve Euler-Lagrange equation using the gradient descent method (GDM). In early days geometric active contour model [25, 26] was computed in the variational framework, a PDE with reference to curve evolution was generated by the Euler-Lagrange equation. Next this PDE was transformed into the PDE for level set function evolution. Afterwards, the energy functional can be directly defined using level sets, then the segmentation task is implemented through minimizing the energy functional. This is

known as the variational level set method. The level set method proposed by Osher and Sethian [123] was originally applied in the field of fluid mechanics. As an effective representation for evolving curves and surfaces because of automatic change of topology, level set method was widely used in scientific researches and engineering practice. In 1993, Caselles et al. [25] took the lead in applying level set method for image segmentation. Since then many more researches began to focus on level sets based segmentation techniques.

Variational level set method uses 3-D implicit equation to represent the closed curve in image domain. Today this curve evolution based representation has become a classic method in the area of variational methods. The numerical computation of level sets can be accomplished without the parametrization process, which shows obvious advantages in calculation and implementation. The main idea of the level set formulation is to implicitly represent a given interface $\Gamma(t)$ (t represents time) as the zero level set of a Lipschitz continuous function $\varphi (\mathbb{R}^2 \rightarrow \mathbb{R})$. The function φ has the following property and its corresponding graph explanation is shown in Fig. 2-2:

$$\begin{cases} \varphi(x, y, t) > 0, & \text{if } (x, y) \text{ is inside } \Gamma(t) \\ \varphi(x, y, t) = 0, & \text{if } (x, y) \text{ is at } \Gamma(t) \\ \varphi(x, y, t) < 0, & \text{if } (x, y) \text{ is outside } \Gamma(t) \end{cases} . \quad (2.1)$$

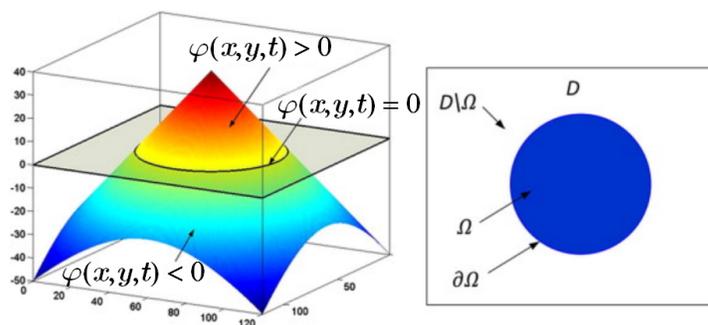


Figure 2-2: Graph explanation for level sets.

In general, φ is defined as a signed distance function in order to keep stability in numerical implementation.

$$\begin{cases} \varphi(x, y, t) = d(\Gamma(t), x, y), & \text{if } (x, y) \text{ is inside } \Gamma(t) \\ \varphi(x, y, t) = 0, & \text{if } (x, y) \text{ is at } \Gamma(t) \\ \varphi(x, y, t) = -d(\Gamma(t), x, y), & \text{if } (x, y) \text{ is outside } \Gamma(t) \end{cases} , \quad (2.2)$$

where $d(\Gamma(t), x, y)$ denotes the shortest Euclidean distance from point (x, y) to $\Gamma(t)$, and from which its Eikona equation is derived as

$$|\nabla\varphi(x, y, t)| = 1. \quad (2.3)$$

For an image defined in domain Ω with boundary $\Gamma(t)$, according to the co-area and area formulas in [5, 102], we will have

$$length(\Gamma) = \int_{\Omega} |\nabla H(\varphi)| dx = \int_{\Omega} \delta(\varphi) |\nabla\varphi| dx, \quad (2.4)$$

$$area(\Omega) = \int_{\Omega} H(\varphi) dx, \quad (2.5)$$

where $H(\varphi)$ and $\delta(\varphi)$ are Heaviside function and Dirac delta function respectively, with the definitions given below

$$H(\varphi) = \begin{cases} 1, & \text{if } \varphi \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (2.6)$$

$$\delta(\varphi) = \frac{\partial H(\varphi)}{\partial \varphi}, \quad (2.7)$$

In fact, these kinds of original definitions for $H(\varphi)$ and $\delta(\varphi)$ are difficult to calculate as there are non-differential functions. In practice, these two functions are often approximated via introducing a small positive parameter ε in the calculation of $H(\varphi)$ and $\delta(\varphi)$ [5, 102] by denoting as $H_{\varepsilon}(\varphi)$ and $\delta_{\varepsilon}(\varphi)$. In this way, we can obtain some feasible computing forms of Heaviside function and Dirac delta function. $H_{1\varepsilon}(\varphi)$ and $\delta_{1\varepsilon}(\varphi)$ were presented in [205] and $H_{2\varepsilon}(\varphi)$ and $\delta_{2\varepsilon}(\varphi)$ were provided in [40, 123, 171] as below:

$$H_{1\varepsilon}(\varphi) = \begin{cases} 1, & \text{if } \varphi > \varepsilon \\ 0, & \text{if } \varphi < -\varepsilon \\ \frac{1}{2}(1 + \frac{\varphi}{\varepsilon} + \frac{1}{\pi} \sin(\frac{\pi\varphi}{\varepsilon})), & \text{if } |\varphi| \leq \varepsilon \end{cases}, \quad (2.8)$$

$$\delta_{1\varepsilon}(\varphi) = \begin{cases} 0, & \text{if } |\varphi| > \varepsilon \\ \frac{1}{2\varepsilon}(1 + \cos(\frac{\pi\varphi}{\varepsilon})), & \text{if } |\varphi| \leq \varepsilon \end{cases}. \quad (2.9)$$

$$H_{2\varepsilon}(\varphi) = \frac{1}{2}\left(1 + \frac{2}{\pi} \arctan\left(\frac{\varphi}{\varepsilon}\right)\right), \quad (2.10)$$

$$\delta_{2\varepsilon}(\varphi) = \frac{1}{\pi} \frac{\varepsilon}{\varepsilon^2 + \varphi^2}. \quad (2.11)$$

Fig. 2-3 draws the graphs of $H_{1\varepsilon}(\varphi)$, $\delta_{1\varepsilon}(\varphi)$, $H_{2\varepsilon}(\varphi)$ and $\delta_{2\varepsilon}(\varphi)$ respectively. In this thesis, we adopt the regularization forms given in $H_{2\varepsilon}(\varphi)$ and $\delta_{2\varepsilon}(\varphi)$.

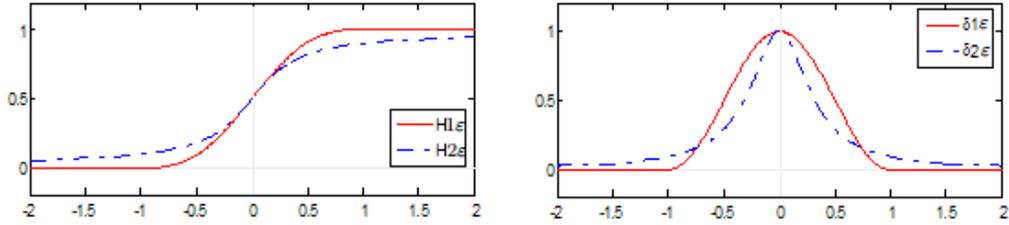


Figure 2-3: Regularization of the Heaviside function and Dirac function.

2.1.3 Mumford-Shah Model

In image segmentation, major advances were made in two-phase image segmentation [37, 40, 41] in the early days. Mumford-Shah model proposed by Mumford D and Shah J [116] is regarded as the most significant region-based model. In recent years, the model has been extended for many other applications. In 2001, Chan and Vese proposed the Chan-Vese model [40] for active contours to detect objects in a given image. It is a numerical realization of the Mumford-Shah model based on variational level sets. Nevertheless, as the complexity of the images increases, two-phase image segmentation was not able to meet the actual needs. Therefore, Mumford-Shah models in different types of multiphase level set frameworks for image segmentation were proposed to satisfy the demands. On the basis of statistical mechanics, Zhao et al. [205] investigated multiphase motion segmentation by using the level set method and proposed a model which can represent n different regions by n level set functions. In order to reduce the number of level set functions, Chan et

al. continued their work and proposed multiphase segmentation model [171] which is a generalization of the Chan-Vese model.

The Mumford-Shah model is not built on the gradient method for edge detection which is sensitive to disconnected edges. The basic idea is to find a pair $(u(x, y), \mathcal{C})$ for a given image f , where $u(x, y)$ is a piecewise smooth approximation of the initial image f , and \mathcal{C} refers to the edges between subregions of the image. The Mumford-Shah functional can be expressed as follows

$$E(u, \mathcal{C}) = \int_{\Omega} (u - f)^2 dx + \mu \int_{\Omega \setminus \mathcal{C}} |\nabla u|^2 dx + \nu \cdot \text{length}(\mathcal{C}), \quad (2.12)$$

where $\Omega \in \mathbb{R}^2$ is the image domain, μ and ν are nonnegative constants. The main aim is to detect the real edges \mathcal{C}_0 , which is achieved by minimizing the functional $E(u, \mathcal{C})$ in (2.12) over u and \mathcal{C} . In the process of energy functional minimization, the proximity between u and f can be guaranteed by the first term of (2.12). The second term keeps the smoothness of u in subregions except edges \mathcal{C} (i.e. $\Omega \setminus \mathcal{C}$), which could make the segmentation curves unaffected by the influences of image noises or textures. The third length term provides a constraint on the evolution curves to match the real object boundaries as close as possible.

However, there are several difficulties in solving Mumford-Shah functional $E(u, \mathcal{C})$ presented in (2.12). It can be observed from the first two terms that u should be obtained on the image domain $\Omega \in \mathbb{R}^2$, while the third term denotes the segmentation curves \mathcal{C} as one-dimensional. In addition, the functional $E(u, \mathcal{C})$ is not convex, and may have multiple local minima. It is hence inevitable that difficulties arise in practical applications. The alternative solutions tackling this problem are to adopt approximate representations. One of the most famous approximate representations is proposed by Chan and Vese [40] based on variational level sets. They transformed the Mumford-Shah functional into the following expression using piecewise constant level set method:

$$\begin{aligned} E(\varphi, c) &= \alpha_1 \int_{\Omega} (c_1 - f)^2 H(\varphi) dx + \alpha_2 \int_{\Omega} (c_2 - f)^2 (1 - H(\varphi)) dx \\ &+ \gamma \int_{\Omega} |\nabla H(\varphi)| dx, \end{aligned} \quad (2.13)$$

where $c = c_1, c_2$ are the image average values of the subregions inside and outside the evolution curve \mathcal{C} .

2.1.4 Euler’s Elastica Constraint

Similar to the previous definitions, Ω is generally supposed to be the image domain and f is the initial image. As described in [160], when performing different image tasks, different definitions for the subset Γ of Ω and f should be introduced. For image denoising [50,211], f represents the noisy image and $\Gamma = \Omega$. For image inpainting [159], we have a damaged area that need to be inpainted: $D \subset \Omega$ denotes the missing or broken part of the image. Here f is the damaged image and $\Gamma = \Omega \setminus D$ is the non-damaged area. Inpainting needs to generate the missing content relying on the available data from Γ , which is achieved by minimizing the corresponding Euler’s elastica energy functional. For image zooming [160], if the size of the given image f is $M_1 \times M_2$, then Ω should be defined on the region size $[r(M_1 - 1) + 1] \times [r(M_2 - 1) + 1]$ (r is a constant scaling factor) and Γ should be defined as $\Gamma = \{(i, j) \in \Omega | i \equiv 1 \text{ mod } r, j \equiv 1 \text{ mod } r\}$. In this case, the intensity values of image on $\Omega \setminus \Gamma$ are interpolated via Euler’s elastica energy. The general Euler’s elastica energy functional for the image tasks mentioned above is given as follows

$$E(\varphi, c) = \int_{\Omega} (\alpha + \beta(\nabla \cdot \frac{\nabla u}{|\nabla u|})^2) |\nabla u| dx + \frac{\eta}{s} \int_{\Gamma} |u - f|^s dx, \quad (2.14)$$

where parameters $\eta > 0$ and $s \geq 1$. Fig. 2-4 presents an example of image denoising based on Euler’s elastica constraint. Fig. 2-4(a) shows a noisy image with Gaussian white noise (mean 0 and the standard deviation 10). The restored images using the classic TV model [140] and Euler’s elastica constraint are presented in (b) and (c) respectively. The result by Euler’s elastica was initially presented in [160]. Some advantages of Euler’s elastica constraint are shown by the comparison between the results shown in (b) and (c): the degraded regions can be recovered better and more image details such as edges can be preserved in (c). More discussions in terms of these properties can be found in [209].

Furthermore, Euler’s elastica constraint can also help to produce good results for image segmentation. Euler’s elastica-based segmentation techniques play an important role due to their crucial roles in boundary reconstruction and image restoration.

(1) Two-phase Euler’s elastica based segmentation: Zhu, Tai, and Chan [212] proposed the Chan-Vese-Euler (CVE) model designed for the foreground shape recovery problem by combining the CV model [40] and Euler’s elastica regularizer [159], actually embedding the properties of inpainting or interpolation contained in Euler’s elastica constraint

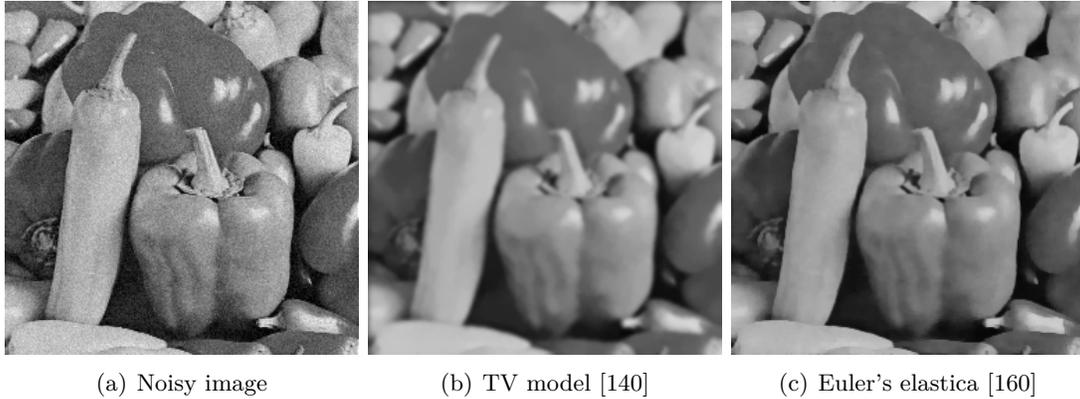


Figure 2-4: Euler's elastica used for image denoising.

into curve evolution. This model could recover the illusory contours and form a complete meaningful object, even without requiring initialization of fixed points. According to their work, the energy functional is defined as

$$\begin{aligned}
 E(\phi, c) &= \alpha_1 \int_{\Omega} (f - c_1)^2 H_{\varepsilon}(\phi) dx + \alpha_2 \int_{\Omega} (f - c_2)^2 (1 - H_{\varepsilon}(\phi)) dx \quad (2.15) \\
 &+ \int_{\Omega} \left(\alpha + \beta \left(\nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right)^2 \right) |\nabla H_{\varepsilon}(\phi)| dx,
 \end{aligned}$$

$$\text{s.t.} \quad |\nabla \phi| = 1,$$

where α_1 , α_2 , α , β are positive penalty parameters, ϕ is a level set representation as described in (2.2). The last term of this functional is the classic Euler's elastica term. We illustrate the effects of Euler's elastica constraint for image segmentation in Fig. 2-5. Fig. 2-5(a) gives the original image with disconnected part, the final results obtained by minimizing the classic segmentation functional [40] and Euler's elastica segmentation functional are presented in (b) and (c) respectively. (d)-(g) show the intermediate steps of recovering the gaps to get an integrated "bar" via Euler's elastica.

(2) Segmentation with depth information: In [121], Nitzberg, Mumford and Shiota defined the problem of segmentation with depth information as a problem of recovering occluded shapes and their ordering relations based on a 2D image. The variables defined in this problem are in three folds: 1) the shapes of the regions R_1, R_2, \dots, R_n to which different objects belong; 2) the ordering relations among objects; 3) the pixel intensities of

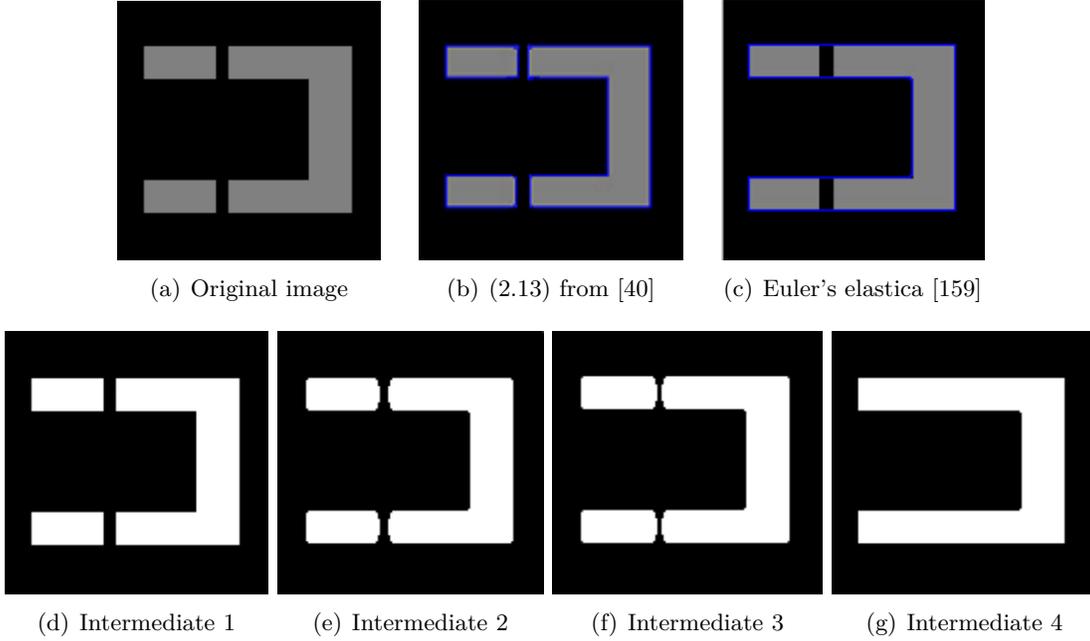


Figure 2-5: Euler's elastica used for image segmentation.

objects. Without loss of generality, one can assume that the objects R_1, R_2, \dots, R_n in an image are in ascending order, i. e., R_1 is the nearest object to the observer while R_n is the farthest one (i.e. background). Let R'_i be defined as the visible part of R_i , i.e., $R'_1 = R_1$, $R'_i = R_i - \bigcup_{j < i} R_j$, ($i = 2, \dots, n$). In addition, $R'_{n+1} = \Omega - \bigcup_{j < n+1} R_j$ is defined as the visible background. Based on the above assumptions and definitions, the level set based energy functional is formulated as in [210]

$$\begin{aligned}
 E(\varphi, c) &= \sum_{i=1}^n \int_{\Omega} (\alpha + \beta |\kappa_i|) |\nabla \varphi_i| \delta(\varphi_i) dx + \int_{\Omega} (f - c_{n+1})^2 \prod_{j=1}^n (1 - H(\varphi_j)) dx \\
 &+ \sum_{i=1}^n \left(\int_{\Omega} (f - c_i)^2 H(\varphi_i) \prod_{j=1}^{i-1} (1 - H(\varphi_j)) dx \right), \tag{2.16}
 \end{aligned}$$

where α, β are two positive penalty parameters, $c_i \in R_i$ is the pixel intensity of the i -th object, and f is the image to be processed. κ_i denotes the curvature of boundary for region R_i . Here $|\kappa|$ is chosen to substitute the square in Euler's elastica term so that the object corners can be preserved when $|\kappa|$ becomes large. In contrast, the model containing curvature related terms in quadratic form would inevitably smear those corners. The difference on the selection of curvature function between κ^2 and $|\kappa|$ for elastica term reflecting in segmenta-

tion performance were also discussed in [166, 210]. The level set function φ is represented by a continuous signed distance function. $H(x)$ and $\delta(x)$ are Heaviside function and Dirac delta function described in detail in [40, 171]. Fig. 2-6 shows the effects of Euler's elastica based segmentation with depth, which can successfully segment objects with occlusions (a hand is occluded by a pen). The original image is given in Fig. 2-6(a), results obtained by traditional multiphase segmentation and Euler's elastica based segmentation with depth approach are shown in (b) and (c). The images used here were directly taken from [166].

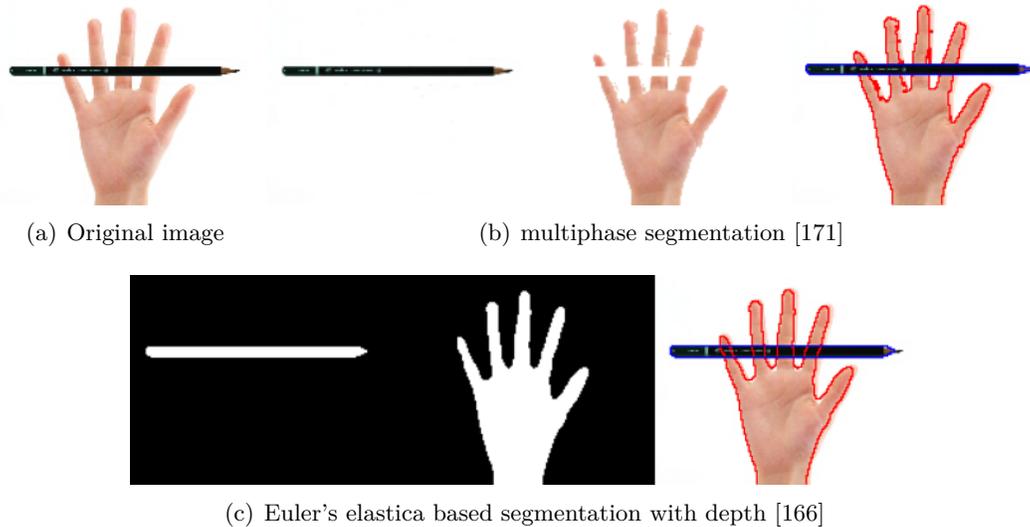


Figure 2-6: Euler's elastica used for segmentation with depth.

2.1.5 Variational Image Processing

Image Recovery

The classical first order Total Variation (TV) model [9] for image de-noising can be stated as the minimization of the following functional

$$E(u) = \int_{\Omega} |\nabla u| dx + \lambda \int_{\Omega} (u - f)^2 dx. \quad (2.17)$$

where $f(x)$ is an observed image defined on the domain, u is the restored image from $f(x)$, " ∇ " is the gradient operator. The first term in (2.17) is the regularization term in total variation form, the second one is the data-fitting term in ℓ_2 form, which is why (2.17) is called the TV- ℓ_2 model.

The idea of replacing the ℓ_2 norm with ℓ_1 norm in the one-dimensional space was first proposed in [3]. The authors [119] extended model (2.17) for image restoration using the following energy functional

$$E(u) = \int_{\Omega} |\nabla u| dx + \lambda \int_{\Omega} |u - f| dx. \quad (2.18)$$

This is the functional of the TV- ℓ_1 model. In comparison with the classic TV model, The TV- ℓ_1 model has the merit of being contrast and geometry preserving meanwhile inheriting the feature of edge preserving.

The counterpart of (2.17), i.e. the MTV- ℓ_2 model for color images using coupled regularization terms was proposed in [17, 189] as follows:

$$E(\vec{u}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |\nabla u_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} (u_i - f_i)^2 dx, \quad (2.19)$$

where $\vec{f} = [f_1, f_2, \dots, f_m]^T$ is a color image defined on domain Ω , and $\vec{u} = [u_1, u_2, \dots, u_m]^T$ is its restored counterpart with $m = 3$. In this case, the MTV- ℓ_1 version is stated as

$$E(\vec{u}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |\nabla u_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} |u_i - f_i| dx. \quad (2.20)$$

The problem of staircase effect still exists for this model. As described in [62, 209], the ℓ_1 -norm of curvature, i.e. total curvature (TC), can be used to design a regularization term for the variation image restoration model in gray image de-noising. The following functional was used in [211]:

$$E(u) = \int_{\Omega} \left| \nabla \cdot \left(\frac{\nabla u}{\sqrt{1 + |\nabla u|^2}} \right) \right| dx + \lambda \int_{\Omega} |u - f| dx, \quad (2.21)$$

where $\nabla \cdot \left(\frac{\nabla u}{\sqrt{1 + |\nabla u|^2}} \right)$ is the mean curvature of the geometric surface $z = u(x)$, and " $\nabla \cdot$ " is the divergence operator. In this chapter, we use another curvature-based form as described below:

$$E(u) = \int_{\Omega} |\kappa| dx + \lambda \int_{\Omega} |u - f| dx. \quad (2.22)$$

where the curvature term κ is represented as $\kappa = \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right)$. According to the total variation term, $|\nabla u|$ can keep piece-wise constant image intensity u , as a result of image smoothing and edges preserving. Similarly, the first term $|\nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right)|$ of (2.22) can preserve the piece-wise constant normal $\frac{\nabla u}{|\nabla u|}$ of images. Thus, the points corresponding to some discontinuous normals, i.e. the corners in images, can be preserved. To the best of our knowledge, the above variational problem cannot be easily solved mathematically since functional (2.22) is non-convex and non-smooth. It has become a difficult issue to find a proper variant equivalent to (2.22) for obtaining its solution.

Illusory Contour Reconstruction

In this subsection, we will review the original variational level set model for illusory contour capture model before presenting our contributions in the next Sections. In [208], Zhu et al. proposed an energy functional as follows. The reconstructed contour can be obtained by minimizing the following energy functional with respect to φ .

$$\begin{aligned}
 E(\varphi) = & \int_{\Omega} (1 + \mu C(d)\kappa^+(d)) |d| \delta(\varphi) |\nabla \varphi| dx + \lambda \int_{\Omega} H(d) H(\varphi) dx \\
 & + \int_{\Omega} (a + b\kappa(\varphi)^2) \delta(\varphi) |\nabla \varphi| dx.
 \end{aligned} \tag{2.23}$$

where Ω represents the image region, μ , λ , a and b are non-negative parameters. $C(x)$ denotes a cut-off function used to limit the calculation to the region near the corners containing concave and convex points and is given by

$$C(x) = \begin{cases} 1, & |x| \leq \varepsilon \\ 0, & |x| > \varepsilon \end{cases}, \tag{2.24}$$

where ε is a positive constant, and we find through experiments that generally the value $\varepsilon = 3$ can produce good results. d and φ in Eq. 2.23 are both level set functions. The zero level set of d represents boundaries of the objects in an image, which is an important reference and control term; and the zero level set of φ represents illusory contours. The signed distance function is usually used as the level set function. As described in [208], a standard signed distance function can be obtained by solving the following Hamilton-Jacobi

equation

$$\frac{\partial d}{\partial t} = \text{sign}(u)(1 - |\nabla d|), \quad (2.25)$$

where u denotes the entire image region, which only takes two distinct values (1 for occupied regions in the image and -1 for the background). Thus, the signed distance function d obtained by (2.25) is positive in the regions occupied by objects and negative on the background. In addition, d can be initialized by standard segmentation methods directly, while the signed distance function φ can be initialized randomly without any constraint.

$H(x)$ and $\delta(x)$ in (2.23) are Heaviside function and Dirac delta function respectively. To avoid the non-differential functions, these two functions are often approximated via introducing a small positive parameter as in [102]. In this way, some feasible forms of Heaviside function and Dirac delta function can be obtained. More details can be found in [40, 171].

More importantly, the curvature κ is represented as $\kappa(\varphi) = \nabla \cdot \left(\frac{\nabla \varphi}{|\nabla \varphi|} \right)$. In [208] κ^2 is used rather than $|\kappa|$ in (2.23) since $|\kappa|$ may not complete the illusory contour in a smooth way. Specifically, $\kappa^+(d) = \max(\kappa(d), 0)$ is used to differentiate the concave points ($\kappa(d) > 0$) and convex points ($\kappa(d) < 0$). Thus in the process of the contour evolution, the concave points can be kept to reconstruct the illusory boundaries. This approach can be seen as the contour completion based on the known concave points.

Image Segmentation

The classic variational formulation of the Chan-Vese model for image segmentation will be introduced here. The Chan-Vese model [40] initially proposed for two-phase image segmentation can be solved as a minimization problem of Mumford-Shah functional [116] which has been transformed into an equivalent variational level set formulations. Based on the piece-wise constant approximation, it divides one image into two heterogeneous regions by using the zero level set. Combined with the binary level set function, the Chan-Vese model can be written as the following form:

$$E(u, \phi) = \alpha_1 \int_{\Omega} (u_1 - f)^2 \phi dx + \alpha_2 \int_{\Omega} (u_2 - f)^2 (1 - \phi) dx + \gamma \int_{\Omega} |\nabla \phi| dx \quad (2.26)$$

where $f(x)$ is a given image, and α_1, α_2 are positive parameters. γ is also a positive parameter which controls the trade-off between the fitting data term and the regularization [8]. $u_1, u_2 \in u$ are piece-wise constants which represent the mean intensity values of the regions. $\phi(x)$ is the binary level set representation supposed to take on either 0 or 1, which is defined as:

$$\phi(x) = \begin{cases} 1, & \text{if } x \in \Omega_1 \\ 0, & \text{if } x \notin \Omega_1, \end{cases} \quad (2.27)$$

where Ω_1 is a closed region inside Ω . This method does not have to reinitialize the level sets during the evolution procedure. Thus the convergence speed can be accelerated. The Chan-Vese model (2.26) can be transformed into the following minimization problem mathematically:

$$\min_{u, \phi \in \{0,1\}} \left\{ E(u, \phi) = \int_{\Omega} r(u_1, u_2) \phi dx + \gamma \int_{\Omega} |\nabla \phi| dx \right\}, \quad (2.28)$$

where $r(u_1, u_2) = \alpha_1(u_1 - f)^2 - \alpha_2(u_2 - f)^2$. Note that the binary constraint for ϕ will cause non-convexity in sub problems (2.28). Authors in [20] demonstrated that certain non-convex minimization problems could be equivalent to the following convex minimization problems by adopting the convex relaxation and the thresholding method. The minimization problem (2.28) can be rewritten as:

$$\min_{u, \phi \in [0,1]} \left\{ E(u, \phi) = \int_{\Omega} r(u_1, u_2) \phi dx + \gamma \int_{\Omega} |\nabla \phi| dx \right\}. \quad (2.29)$$

Recently the Chan-Vese model has become an important tool for image segmentation. In this model, active contours can automatically detect interior objects in a given image whose boundaries are not necessarily defined using techniques of curve evolution. Moreover, the initial curve can be placed anywhere in the image. It can detect and preserve the locations of boundaries well without smoothing the initial image, even if the image is noisy.

Nonlinear diffusion regularization

The main idea of the non-linear diffusion technology is that the diffusion coefficient depends on the local image property in the diffusion procedure. Specifically, the diffusion coefficient

can automatically increase in smooth regions so that the noise can be suppressed. Similarly, the coefficient can automatically decrease in edge regions so that the edges can be preserved. After Perona and Malik proposed the PM model [128], many extensions are presented. In 1997, its general variational model [9] was proposed by Aubert and Vese. Several customarily used potential functions (PF) φ are given in Table 2.1.

Table 2.1: Potential functions for the regularization term.

No.	$\varphi(\nabla\phi)$	source
(i)	$ \nabla\phi ^p, 0 < p \leq 2$	[119]
(ii)	$\sqrt{1 + \nabla\phi ^2}$	[174]
(iii)	$\sqrt{1 + \nabla\phi ^2} - 1$	[1]
(iv)	$\frac{ \nabla\phi ^2}{1 + \nabla\phi ^2}$	[60]
(v)	$\log(1 + \nabla\phi ^2)$	[71]
(vi)	$\log(\cosh(\nabla\phi))$	[43]
(vii)	$1 - \lambda^2 e^{-\frac{ \nabla\phi ^2}{2\lambda^2}}$	[120]
(viii)	$\lambda^2 \log(1 + \frac{ \nabla\phi ^2}{\lambda^2})$	[120]
(ix)	$2\lambda^2(\sqrt{1 + \frac{ \nabla\phi ^2}{\lambda^2}} - 1)$	[42]
(x)	$ \nabla\phi - \alpha \log(1 + \frac{ \nabla\phi }{\alpha})$	[66]

Nonlinear diffusion can transform the image estimation into a temporal evolution whose stable solution is the favourable result of image processing. It can produce new filtering methods by simply recombining some other filtering methods. Due to its strong robustness, it can be extended into many other aspects in image processing.

Segmentation Models Incorporating Noise Distributions

Noisy image segmentation problems are investigated [108, 127, 146] using specific parameter estimation based on different noise distributions. All the related parameters are calculated via the maximum a posteriori probability (MAP) estimation from the viewpoint of Bayesian probability models. For example, estimation of variance information is used for images degraded with Gaussian noise; The square of image intensity value with capacity of enhancing weak properties is incorporated in the Rayleigh model; Models with great segmentation performance of dealing with Poisson and Gamma noises are built on the standard deviation

and average. The general variational functional is written as follows

$$E(\theta, \phi) = \alpha_1 \int_{\Omega} Q_1(x, \theta_1) \phi dx + \alpha_2 \int_{\Omega} Q_2(x, \theta_2) (1 - \phi) dx + \gamma \int_{\Omega} |\nabla \phi| dx, \quad (2.30)$$

where $\alpha_1, \alpha_2, \gamma$ are positive penalty parameters, ϕ is a binary level set as defined in functional (2.15). Specific representations of function Q derived from the maximum likelihood method and the computation of their related parameters are given in Table 2.2. $\theta = (\mu, \sigma)$ refers to the corresponding parameters of function Q need to be estimated.

Table 2.2: Potential functions of different noise distributions.

Functions	Gaussian noise	Rayleigh noise
$Q_{i(i=1,2)}$	$\frac{1}{2} \log 2\pi + \log \sigma_i + \frac{(f - \mu_i)^2}{2\sigma_i^2}$	$2 \log \sigma_i - \log f + \frac{f^2}{2\sigma_i^2}$
Parameters $\theta_i = (\mu_i, \sigma_i)$	$\mu_i = \frac{\int_{\Omega} f \phi^{2-i} (1-\phi)^{i-1} dx}{\int_{\Omega} \phi^{2-i} (1-\phi)^{i-1} dx}$ $\sigma_i^2 = \frac{\int_{\Omega} (f - \mu_i)^2 \phi^{2-i} (1-\phi)^{i-1} dx}{\int_{\Omega} \phi^{2-i} (1-\phi)^{i-1} dx}$	$\sigma_i^2 = \frac{\int_{\Omega} f^2 \phi^{2-i} (1-\phi)^{i-1} dx}{2 \int_{\Omega} \phi^{2-i} (1-\phi)^{i-1} dx}$
Functions	Poisson noise	Gamma noise
$Q_{i(i=1,2)}$	$\sigma_i - f \log \sigma_i$	$\frac{f}{\mu_i} + \log \mu_i$
Parameters $\theta_i = (\mu_i, \sigma_i)$	$\sigma_i = \frac{\int_{\Omega} f \phi^{2-i} (1-\phi)^{i-1} dx}{\int_{\Omega} \phi^{2-i} (1-\phi)^{i-1} dx}$	$\mu_i = \frac{\int_{\Omega} f \phi^{2-i} (1-\phi)^{i-1} dx}{\int_{\Omega} \phi^{2-i} (1-\phi)^{i-1} dx}$

2.2 Stochastic Programming Theory

Mathematical programming is about decision making, or planning. Stochastic programming is a type of mathematical programming specifically for decision making under uncertainty. In other words, stochastic programming can be regarded as a type of mathematical programming with random parameters. Stochastic programming models utilize the fact that although uncertainty is contained in the optimization problem, probability distributions governing the data should be known or can be estimated. In most cases, the real world problems inevitably include unknown parameters while a decision should be made. The prerequisite for applying stochastic programming is that the parameters are assumed to lie in some given possible values even though they are uncertain. First formulated as mathematical programs in the late 1950s, nowadays stochastic programming has become a useful modelling paradigm. Here we also take some applications of stochastic programming as examples in solving real-life decision problems in Fig. 2-7.

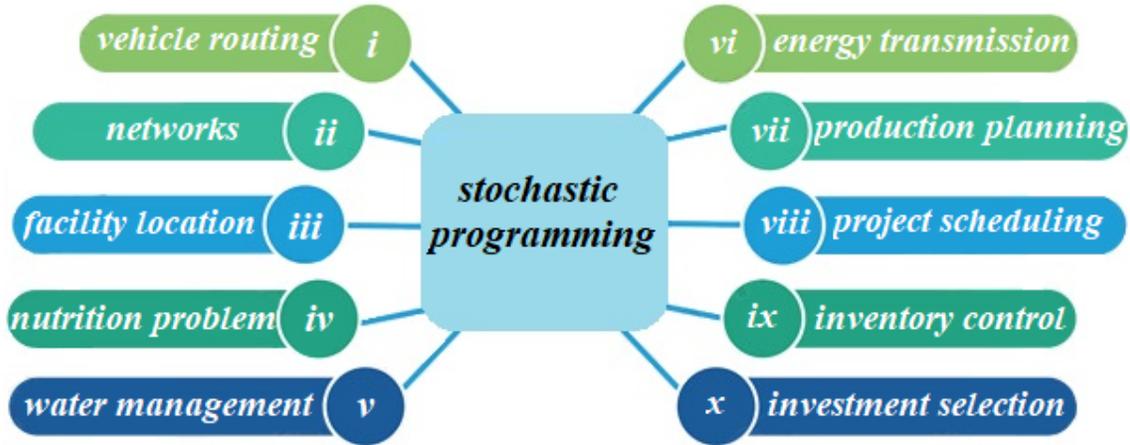


Figure 2-7: The list of application areas of stochastic programming.

2.2.1 Two-stage Stochastic Programming

Two-stage stochastic programming [130, 134, 150] is well known as the most widely studied and applied stochastic programming model. All the random event decisions should be made in its first stage. Then a final decision is made in the second stage that compensates for any possible bad result from the first-stage decision. With the same definition as the stochastic programming, a two-stage stochastic programming is also an optimization problem which is determined by the two decisions from each stage individually. In the first stage, all variables associated with every random vector need to be decided before the realization of uncertain vector, whereas variables for the second stage are decided after observing the random vector. Let ξ be the random vector and let Ξ be the set of all the possible realizations of ξ (assumed to be finite). Then, the response mapping $x(\cdot)$ of a two-stage stochastic programming has the following form:

$$x(\cdot) : \xi \mapsto x(\xi) = (x_1(\xi), x_2(\xi)) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} = \mathbb{R}^n,$$

where $x_1(\xi)$ is the first-stage decision variable and $x_2(\xi)$ is the second-stage decision variable. Consider the Hilbert space \mathcal{L} consisting of all the mappings $x(\cdot) \in \mathbb{R}^n$. Since in the two-stage model the first decision is made before ξ is realized, the solution to the two-stage stochastic programming problem should satisfy the so-called nonanticipativity constraint

$x(\cdot) \in \mathcal{N}$, where

$$\mathcal{N} := \{x(\cdot) = (x_1(\cdot), x_2(\cdot)) \in \mathcal{L} \mid x_1(\xi) \text{ does not depend on } \xi\}.$$

Clearly, \mathcal{N} is a subspace of \mathcal{L} . In addition, suppose every decision $x(\cdot)$ must satisfy a set of constraints and the constraints generally depend on ξ . We write this fact in the form of

$$x(\cdot) \in \mathcal{C} \subset \mathcal{L}, \text{ which means } x(\xi) \in C(\xi) \forall \xi \in \Xi,$$

where each $C(\xi)$ refers to a nonempty closed convex subset of \mathbb{R}^n and the set \mathcal{C} therefore denotes a nonempty closed convex subset of \mathcal{L} . Ξ is the set of all the possible realizations of ξ (assumed to be finite). Let the total cost of the decision $x(\xi)$ in two stages be $f(x(\xi), \xi)$. Then the two-stage stochastic programming problem can be formulated as

$$\begin{aligned} \min_{x(\cdot)} \quad & F(x(\cdot)) = \mathbb{E}_\xi[f(x(\xi), \xi)] := \sum_{\xi} p(\xi) f(x(\xi), \xi) \\ \text{s.t.} \quad & x(\cdot) \in \mathcal{C} \cap \mathcal{N}, \end{aligned} \tag{2.31}$$

where \mathbb{E} stands for the expectation, $p(\xi) > 0$ represents the known probability of each scenario ξ , and the sum of these probabilities is 1, that is: $\sum_{\xi} p(\xi) = 1$. In the formulation shown in (2.31), in the first stage we have to make a decision x before the realization of the uncertain data ξ . In the second stage, after ξ becomes available, we can obtain the optimum by solving an appropriate optimization problem. Fig. 2-8 demonstrates the procedure of two-stage stochastic programming.

2.2.2 The Progressive Hedging Algorithm for Two-stage Stochastic Programming

The progressive hedging algorithm (PHA) was originally designed by Rockafellar and Wets [136] for multistage stochastic minimization problems and it has recently been extended [134] to monotone stochastic variational inequality (SVI) problems of the form

$$x(\cdot) \in \mathcal{N} \text{ and } w(\cdot) \in \mathcal{M} \text{ such that } -\nabla F(x(\cdot)) - w(\cdot) \in N_{\mathcal{C}}(x(\cdot)), \tag{2.32}$$

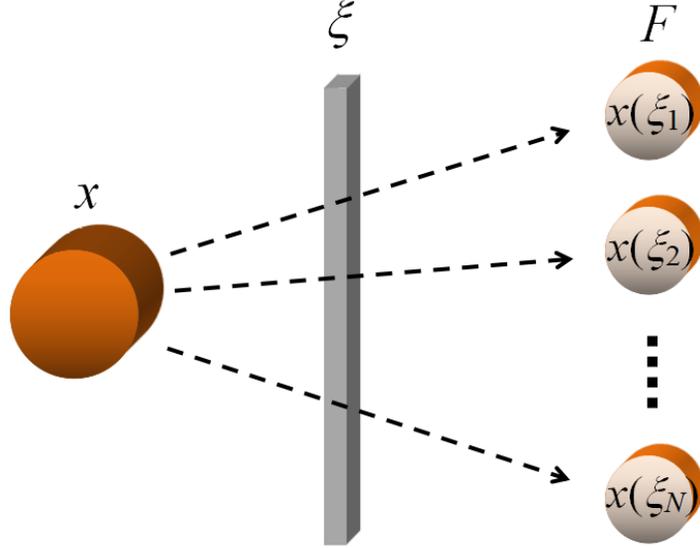


Figure 2-8: A diagram for two-stage stochastic programming.

where F is a convex continuously differentiable functional, $N_{\mathcal{C}}(x(\cdot))$ is the normal cone to \mathcal{C} at $x(\cdot)$, \mathcal{N} is a linear subspace of the Hilbert space \mathcal{L} composed of all $x(\cdot)$, and \mathcal{M} is the complementary subspace of \mathcal{N} . The notation of $x(\cdot)$ indicates that the solution of Problem (2.32) is a function and \mathcal{L} is therefore a functional space. This understanding of (2.32) is particularly suitable for multistage stochastic optimization, where the best response function $x(\xi)$ is sought for each realization of a random vector ξ . In such cases Problem (2.32) represents the necessary condition for optimality and it is also sufficient if the stochastic programming problem is convex.

If $f(\cdot, \xi)$ in (2.31) is a continuously differentiable convex function, the following Karush-Kuhn-Tucker (KKT) conditions will be both necessary and sufficient for the optimality of $x(\cdot)$:

$$-\nabla F(x(\cdot)) \in N_{\mathcal{C} \cap \mathcal{N}}(x(\cdot)). \quad (2.33)$$

When certain constraint qualifications such as $\text{ri } \mathcal{C} \cap \mathcal{N} \neq \emptyset$ (or $\mathcal{C} \cap \mathcal{N} \neq \emptyset$ if the sets $\mathcal{C}(\xi)$ are all polyhedral) are met, the following equality holds.

$$N_{\mathcal{C} \cap \mathcal{N}}(x(\cdot)) = N_{\mathcal{C}}(x(\cdot)) + N_{\mathcal{N}}(x(\cdot)).$$

Since \mathcal{N} is a subspace, which yields that $N_{\mathcal{N}}(x(\cdot)) = \mathcal{N}^\perp := \mathcal{M}$. Based on this equality, the SVI (2.33) can be rewritten as the following equivalent form:

$$x(\cdot) \in \mathcal{N} \text{ and } w(\cdot) \in \mathcal{M} \text{ such that } -\nabla F(x(\cdot)) \in N_{\mathcal{C}}(x(\cdot)) + w(\cdot). \quad (2.34)$$

Subsequently, SVI (2.32) can be easily obtained from (2.34) just through moving the term $w(\cdot)$ from the righthand side to the lefthand side. One can also regard SVI (2.32) as an equivalent alternative extension of (2.33). The focus of PHA is to transform solving the original problem stated in (2.33) into solving the problem

$$-\nabla F(x(\cdot)) - w(\cdot) \in N_{\mathcal{C}}(x(\cdot)) \iff -\nabla_x f(x(\xi), \xi) - w(\xi) \in N_{C(\xi)}(x(\xi)), \forall \xi. \quad (2.35)$$

As described in [134], the powerful feature of PHA is its capability of decomposition in terms of solving separate "deterministic" problem for each scenario ξ , while keeping each iteration in the space \mathcal{N} by projection. In this regard, the vector $w(\cdot)$ serves as the multiplier vector for the nonanticipativity constraint, each of the w -iteration is also kept in the space \mathcal{M} , as shown in the following description of the algorithm.

Algorithm 2.1 PHA in two-stage stochastic programming

Given $x^\nu(\cdot) \in \mathcal{N}$ and $w^\nu(\cdot) \in \mathcal{M} := \mathcal{N}^\perp$ with a fixed parameter $r > 0$,

Step 1. For every ξ , solve

$$\hat{x}^\nu(\xi) = \arg \min_{x(\xi) \in C(\xi)} \{f(x(\xi), \xi) + w^\nu(\xi)^T x(\xi) + \frac{r}{2} \|x(\xi) - x^\nu(\xi)\|^2\};$$

Step 2. Update $x^{\nu+1}(\cdot) = P_{\mathcal{N}}(\hat{x}^\nu(\cdot))$, $w^{\nu+1}(\cdot) = w^\nu(\cdot) + r(\hat{x}^\nu(\cdot) - x^{\nu+1}(\cdot))$.

$\nu := \nu + 1$, **repeat** until a stopping criterion is met.

Here the existence and uniqueness of the solution $\hat{x}^\nu(\xi)$ will be guaranteed by the fact that the r -term makes the related minimization subproblems be strongly convex. $P_{\mathcal{N}}$ is the projection mappings onto the linear subspaces \mathcal{N} , and actually has the following explicit form in the two-stage stochastic programming:

$$P_{\mathcal{N}}(x(\cdot)) := (z_1(\cdot), z_2(\cdot)) \text{ with } z_1(\xi) = \sum_{\xi} p(\xi) x_1(\xi) \text{ and } z_2(\xi) = x_2(\xi) \forall \xi. \quad (2.36)$$

If problem (2.31) is convex and there exists at least one solution, the sequence generated by

PHA converges to an optimal solution. More details in terms of theorems and proofs can be found in [134]. An alternative proof of its convergence is provided in [153] in terms of alternating direction method of multipliers (ADMM). The convergence properties of PHA for the two-stage stochastic programming problem can be summarized as follows.

2.2.3 Convergence properties of the PHA

(Theorem 2 of [134]) As long as the (monotone) variational inequality (2.32) satisfies the constraint qualification and has at least one solution, the sequence $\{x^\nu(\cdot), w^\nu(\cdot)\}$ generated by the progressive algorithm will converge to some pair $(\bar{x}(\cdot), \bar{w}(\cdot))$ satisfying (2.34) and thus furnish $\bar{x}(\cdot)$ as a solution to (2.32). In this case, the distance expression

$$\|x^\nu(\cdot) - \bar{x}(\cdot)\|^2 + \frac{1}{r^2} \|w^\nu(\cdot) - \bar{w}(\cdot)\|^2$$

will form a decreasing sequence that goes to zero. The decrease will surely be at a linear rate if, in particular, the sets $C(\xi)$ are polyhedral and the functions $f(x(\xi), \xi)$ are convex quadratic (including linear as a special case).

2.3 Deep Learning models

Deep learning techniques are derived from a broader set of machine learning methods based on artificial neural networks (ANNs). ANNs were computing systems established according to the structure and action of the human brain. Deep learning often performs better in feature representation by learning from large amounts of data.

There are three main learning types in conducting computer vision tasks: supervised, semi-supervised and unsupervised. Supervised models usually accomplish the learning task based on the available examples which are well labeled and obtain the desired function for predicting correctly. Semi-supervised training does not require as much effort as supervised training to collect the data. Only a small amount of labeled examples as well as a large number of unlabeled or weakly labeled examples are required during training process. Unsupervised learning refers to the system can make predictions for unseen data totally without any prior labelled samples.

2.3.1 Deep Feedforward Network

Deep feedforward network is one typical deep learning model, which is also known as the multilayer perceptron (MLP). An example of a simple neuron is shown in Fig. 2-9. In a neuron, input information $\mathbf{x} = [x_1, \dots, x_i, \dots, x_n]$ flows to the output o through the function $f(\cdot)$ in the forward direction. Such an architecture consisting of a single neuron with one or more inputs, a processor, and a single output is called a perceptron. In Fig. 2-9, a mapping is defined as

$$\begin{cases} s = \sum_{i=1}^n w_i x_i \\ o = f(s) \end{cases} \implies \mathbf{o} = f(\mathbf{x}; \boldsymbol{\theta}), \quad (2.37)$$

where the bias is neglected. The aim of deep feedforward network is to obtain the appropriate approximation of function $f^*(\cdot)$. From the mapping presented in (2.37), the optimal approximation can be obtained by tuning the parameter $\boldsymbol{\theta} = \mathbf{w} = [w_1, \dots, w_i, \dots, w_n]$ with the learning algorithm, rendering $f(\cdot)$ as close as possible to $f^*(\cdot)$.

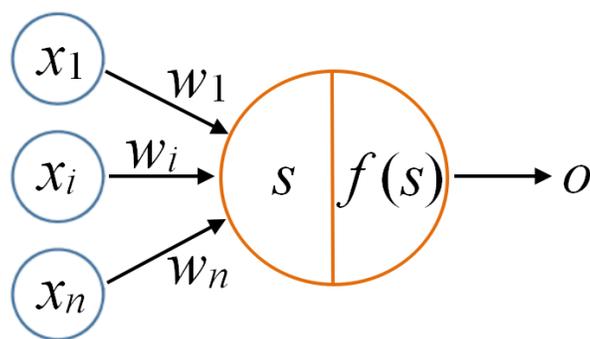


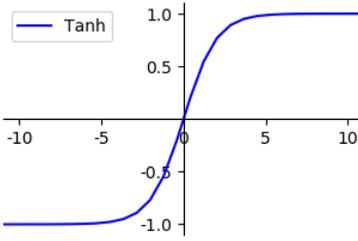
Figure 2-9: An example of one neuron.

In practice, the information feature should be described via nonlinear function. Hence $f(\cdot)$ should be a pointwise nonlinear activation function. There are three options for the common nonlinear activation functions: a hyperbolic tangent (Tanh), logistic sigmoid and the rectified linear (ReLU) function. In general, ReLU is recommended due to the non-saturation of its gradient, which greatly accelerates the convergence of stochastic gradient descent compared to the Tanh or sigmoid functions. Table 2.3 gives the formulations and plots of the three activation functions respectively.

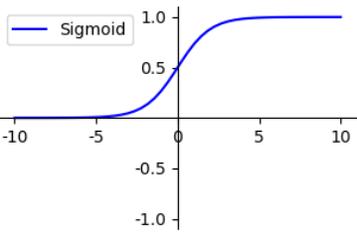
Fig. 2-9 describes the basic architecture of a neuron, and a neural network is composed

Table 2.3: Formulations and plots of different active functions.

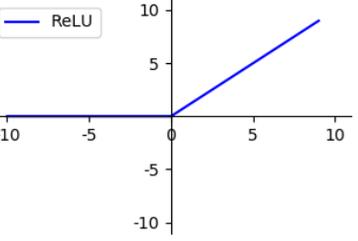
Hyperbolic tangent	Logistic sigmoid	Rectified linear unit
$Tanh(x) = \frac{e^{2x}-1}{e^{2x}+1}$	$sigmoid(x) = \frac{1}{1+e^{-x}}$	$ReLU(x) = max(0, x)$



Tanh



Sigmoid



ReLU

of more than one neurons in most cases. One typical example of deep feedforward network (i.e. multilayer perceptron) or MLP is given in Fig. 2-10. A MLP consists of at least three layers of nodes: an input layer, a hidden layer (may contain many hidden layers) and an output layer. In the MLP example shown in Fig. 2-10, the input layer has four nodes (blue), the output layer contains one node (red) and the middle three are hidden layers (six nodes for each layer). MLP was successfully used in various applications, such as function approximation, classification, pattern recognition and so on. There are many disadvantages of MLP which make it insufficient for modern advanced computer vision tasks. The first disadvantage is that the number of total parameters in MLP may be very high since one perceptron is used for each input, inevitably leading to redundancy in such high dimensions. Another disadvantage is that it is prone to lose pixels and neglect spatial information, which is caused by the inputs in flattened vectors. In recent years, MLP has been surpassed by Convolutional Neural Network (CNN). There are three main reasons for this situation: (1) CNN is able to reduce the number of parameters by sharing weights; (2) The smaller and shared weights make CNN much more efficient in computer vision tasks; (3) Furthermore, the problem of losing pixels and spatial information is also solved by CNN using the sparse connectivity.

2.3.2 Convolutional Neural Networks

Convolutional neural networks (CNNs) [52, 88, 92, 152] can be regarded as a special case of fully connected MLPs, which is designed for processing data that has a grid-like struc-

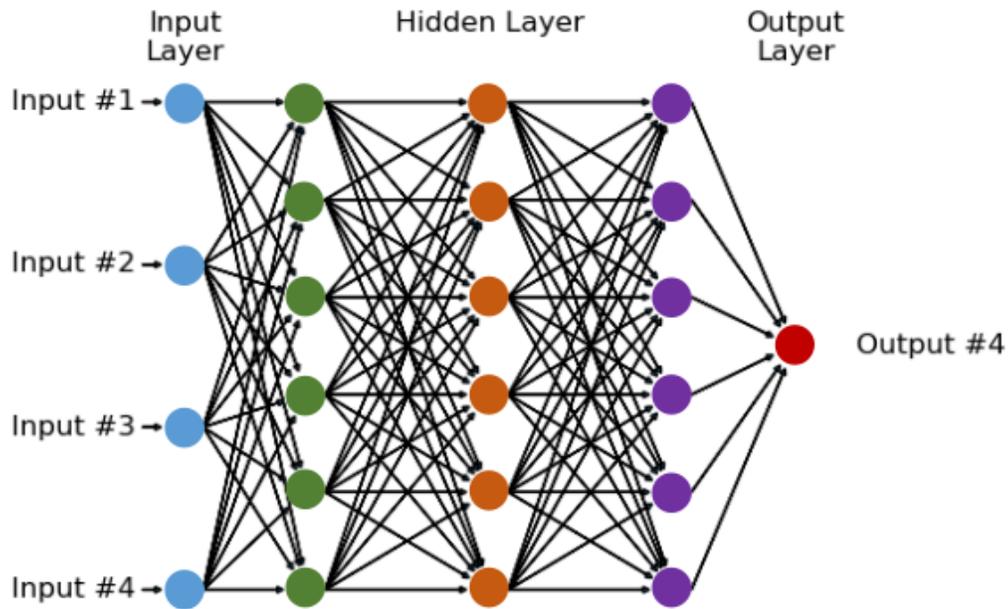


Figure 2-10: An example of multi-layer perceptron (MLP).

ture. For example, time series data is in one dimensional grid structure as the samples taken at regular time intervals and images can be seen as two dimensional pixel grid. The word "convolutional" in convolutional neural networks demonstrates that the convolution operation is used, which is a special type of linear operation. CNN is a class of neural networks that applies convolution operation instead of the general matrix multiplication in at least one layer. An example of a convolution operation is shown in Fig. 2-11. The convolutional layer aims to produce feature representations of the inputs by training and in practice different feature maps are needed to attain the desired performance. The feature maps are generated after the filters or kernels have passed over the image (a feature map is generated by a single filter). The filter can provide important information about the influence of nearby pixels, which forms the basis for image analysis. The size of a filter is usually specified by the user (generally 3×3 or 5×5) and the filter is moved across the image from top left to bottom right. These filters introduce translation invariance and parameter sharing, bringing in tremendous success for CNNs in a variety of applications. When designing the network, the values of the filters are randomly initialized, and then continuously updated through training. It is rare that two filters are the same, unless the number of chosen filters is extremely large. Following convolution operations shown in Fig.

2-11, the value at location $(0, 0)$ of feature map is calculated as follows:

$$o(i_o, j_o) = \sum_{i=0}^m \sum_{j=0}^n x_{i_o+i, j_o+j} w_{i,j}, \quad (2.38)$$

In the above, $o(i_o, j_o)$ is the point value at location (i_o, j_o) of the feature map, and the coordinate range of the filter is $[0 : m, 0 : n]$. $x_{i,j}$ refers to the image intensity and $w(i, j)$ denotes the value of the filter (at location (i, j)) respectively. It should be noted that the equation obtaining $o_{0,0}$ showed in Fig. 2-11 does not follow the strict definition for convolution in theory. Actually, (2.38) is the cross-correlation function. Cross-correlation means sliding filter across an image. Convolution means sliding a flipped kernel across an image. The commutativity of convolution is guaranteed by the flipped kernel. Convolution is usually used together with other functions in practice, and this combination is commonly noncommutative, which means the order of when each function is applied cannot be changed. Hence cross-correlation is directly used as convolution in many machine learning libraries.

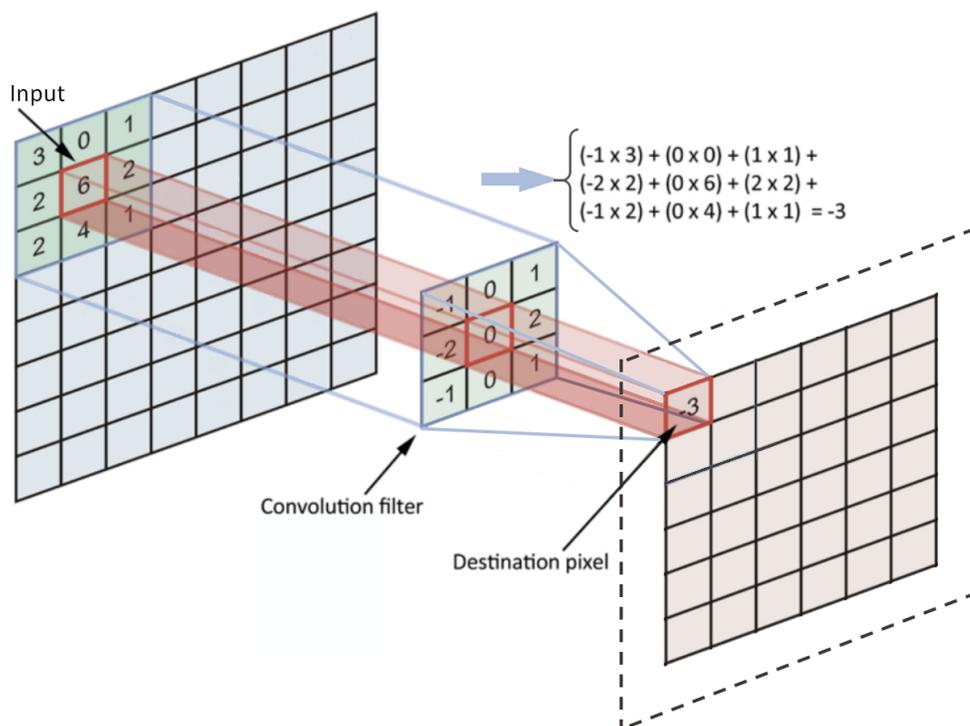


Figure 2-11: An example of convolution operation.

If convolutions are applied on a normal image, the result will be down-sampled by an amount depending on the size of the filter, as shown in Fig. 2-11. After the convolution

operation is performed by sliding the 3×3 filter over the input, the size of the resulting feature map is smaller than the input (the dotted rectangle in Fig. 2-11 represents the original size). This is because the convolution filter needs to be contained in the input. Padding is a very useful tool to make the feature maps the same size as the input image. After feature maps are obtained, they are then taken through an activation function to introduce nonlinearities to CNNs.

CNNs are also in the multi-layer structure that can extract different features. An example of basic CNN architecture is presented in Fig. 2-12 with two convolutional layers, two pooling layers, and two fully connected layers which makes the final predictions of the image for classification.

Convolutional Layers

Convolutional layers are the layers where filters are applied to the previous layer. It aims to learn feature representations of the inputs by convolving the input with kernels and then pass the results to the next layer. Different feature maps can be obtained by using a set of different kernels of the convolutional layer. This is where most of the user-specified parameters are set in the network.

Pooling Layers

The pooling layer (also called the sampling layer) is used to reduce the dimensionality of the network, as well as manage the sensitivity of the filters to noise and variations. This has the effect of making the resulting down-sampled feature maps more robust to changes in the position of the feature in the image. Pooling layer is usually added after the convolutional layer. It can be seen from the example given in Fig. 2-11, the feature map of the first convolutional layer is connected to its corresponding feature map of the pooling layer. Then the output of the last pooling layer is passed to fully connected layers for final predictions. There are two common pooling methods used in the pooling operation: average pooling and max pooling, which calculate the average/maximum value for each patch on the feature map.

Fully Connected Layers

The fully connected layers are placed in the last part of the network before producing the final predictions. This is similar to the output layer of an MLP where every neuron in the previous layer is connected to every one in the next layer. The output of the previous layers is flattened, then is fed into a fully connected layer that drives the final classification decision.

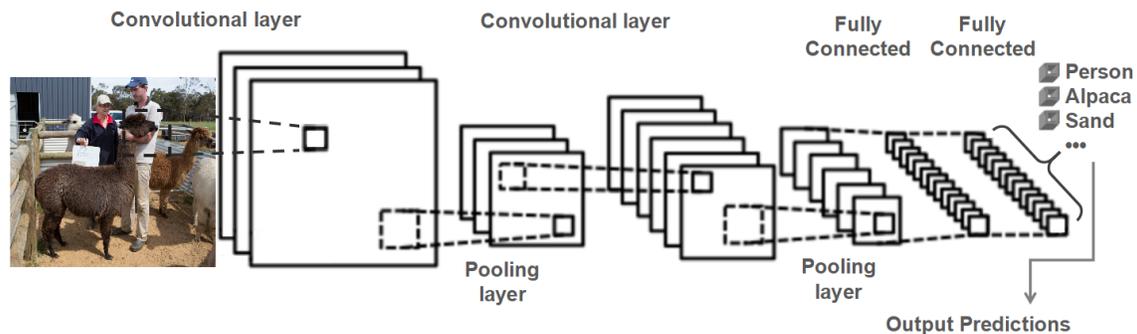


Figure 2-12: An example of basic CNN architecture.

Apart from the above three main types of layers in a CNN, other types such as deconvolution layer (or called a transposed convolutional layer) can also be added into the network design when conducting tasks involving CNN visualization, pixel-wise prediction, image generation, etc. The common way for implementing deconvolution layer is the same as the implementation for convolution layer, which means deconvolution layer with kernel size of $(o, i, k \times r, k \times r)$ equals convolution layer with kernel size of $(o \times r^2, i, k, k)$, r refers to the upscaling ratio. For example, authors in [200] applied deconvolution to unsupervised feature learning by learning a set of kernels and feature maps to reconstruct the image in an unsupervised way. Deconvolution was utilized in [199] to map the feature activities obtained from convolution layer back to the input pixel space, showing what input pattern can result in a given feature map. Since pixel-wise prediction like image segmentation [101] and image generation [131] need to be implemented on the same size of the original image, deconvolution can also be used as a way for upsampling. An example briefly introducing the CNN architecture with deconvolution is given in Fig. 2-13.

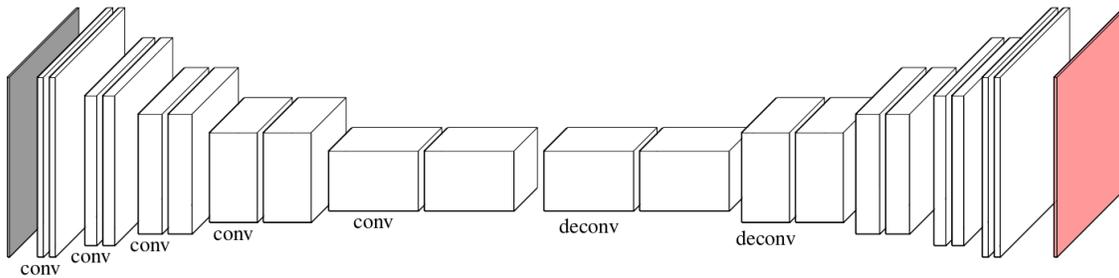


Figure 2-13: An example of CNN architecture with deconvolution.

2.4 Summary

Some basic introductions including preliminary knowledge related to the topics of this thesis are provided in this chapter. First, a brief introduction of some widely used variational techniques is presented: the evolution of PDE, basic operations, definitions of level sets, the original concept of segmentation, and important regularized terms, followed by summaries of classic variational models. Next, we describe the stochastic programming theory shortly. The two-stage stochastic programming of this theory motivates us a lot to explore the impact of randomness of inputs on outputs. Progressive hedging algorithm for solving two-stage stochastic minimization problems as well as the convergence is also demonstrated. Last, some basics of deep learning such as common network architectures including artificial neural networks (ANNs) model, the multilayer neural networks and the convolutional neural networks (CNNs) models are introduced.

After the associated background knowledge have been reviewed in this chapter, the following chapters will describe different approaches tackling various image tasks inspired by variational methods, stochastic programming and deep learning techniques in detail. To begin with, the first focus of this thesis with regard to variational and PDE based methods for image restoration, illusory contour reconstruction and segmentation will be illustrated in the next chapter.

Chapter 3

Variational and PDE Based Methods for Diverse Image Tasks

3.1 Introduction

Image Restoration

It is a crucial step in image analysis for applications such as medical imaging, pattern recognition, and video surveillance, etc. In recent two decades, numerous variational models [8, 124, 126, 148] have been proposed for a very wide range of image processing applications, and the classical first order Total Variation (TV) model or Rudin–Osher–Fatemi (ROF) model [9] can be considered as the starting point of these models due to its excellent edge preserving property. However this TV model has some demerits, such as staircase effects, smearing corners and image contrast reduction. In order to remedy its drawbacks of contrast loss, Alliney [3, 4], Nikolova [119], Chan and Esedoglu [35] replaced the ℓ_2 data fitting term in TV model with the L1 norm. In order to ameliorate the staircase effect, some researchers proposed various higher order variation models. For instance, Lysaker et al. [105] proposed a model that directly incorporates the second order derivative information into the model

⁰© 2018 Elsevier. This chapter is based on, as well as with permission from [Tan, L., Liu, W., & Pan, Z. Color Image Restoration and Inpainting via Multi-Channel Total Curvature. *Applied Mathematical Modelling*, 2018. DOI: <https://doi.org/10.1016/j.apm.2018.04.017>].

⁰© 2019 Springer. This chapter is based on, as well as with permission from [Tan, L., Li, L., Liu, W., & Pan, Z. Fast Computational Approach for Illusory Contour Reconstruction. *Multimedia Tools and Applications*, 2019. DOI: <https://doi.org/10.1007/s11042-018-6546-8>].

⁰© 2019 American Institute of Mathematical Sciences. This chapter is based on, as well as with permission from [Tan, L., Li, L. An, S., & Pan, Z. Nonlinear diffusion based image segmentation using two fast algorithms. *Mathematical Foundations of Computing*, 2019. DOI: <https://doi.org/10.3934/mfc.2019011>].

for image de-noising. Chambolle and Lions [31] used a higher order method by means of an informal-convolution of the first order and second order regularization terms. The former can preserve edges and the latter can keep the image smoothness. Based on the first order ROF model, Bredies et al. [18] proposed a total generalized variation (TGV) model with different order for image de-noising. Another example of combining the first and second order regularization terms were presented by Chan et al. [34], who applied the TV minimization together with weighted coefficients based on Laplacian. Zhu and Chan [209] and Yip and Zhu [192] proposed a mean curvature variation model to effectively remove noises. It is worth mentioning that the curvature-based restoration models (i.e. Zhu and Chans model [209]) are able to preserve edges and corners while eliminating the staircase effect completely. In addition, Chan et al. [151] used the Eulers elastica of level curves as a smooth function in a regularization term for image inpainting. In [38, 56, 110], they used the curvature term directly to solve inpainting problems, producing excellent edge and corner preservation. After the curvature-based models were first proposed by Nitzberg and Mumford [122] in early 1990s, these models are widely extended for segmentation with depth [166, 210] and construction of illusory shapes [83] etc.

Although the aforementioned curvature based models can overcome the side effects of the ROF model effectively, they cannot be directly applied to color image processing due to the different diffusion in each image channel. Technically, the coupled regularization terms for color image restoration must be designed similarly as in the first order variation model for color image restoration. For color image processing, restoring each channel separately using the first order variation models usually results in blurred edges [17, 104, 167] due to the different diffusion in each channel. Considering the coupling effects, Sapiro and Ringach [143] proposed an anisotropic filter to diffuse color image according to directions of maximal and minimal changes in an image. The concept of Riemannian geometry and law of Polyakov action [139] in theoretical physics were used by Brook et al. [22] in their proposed energy minimization models. A celebrated Color TV (CTV) model was proposed in [17] by considering the global coupling of different channels in color image. Yang et al. [189], Bresson and Chan [19], Aujol and Kang [10] and Duval et al. [55] proposed the Multi-channel TV (MTV) and Vector TV (VTV) models, respectively, based only on the local coupling information of different color channels, thus these models have higher computational efficiency than the CTV models. In [196], Yu et al. compared the edge preserving

ability of different vector variation models and tested the efficiency of their corresponding split Bregman algorithms. Moreover, the non-local [81] CTV and MTV regularization terms were also applied to color image denoising and inpainting.

Illusory Contour Reconstruction

Illusory contours [84] in an image are a well-known phenomenon, which contains some missing boundaries of objects, as shown in Fig. 1-1. They are visual organization of disconnected edge segments which are extremely difficult for computers to identify automatically but easy for human to obtain. Illusory contour capture/reconstruction [59, 85, 133] aims to fill the absent boundaries or recover some missing shapes. In fact, illusory contour capture plays an important role in image analysis and computer vision, such as object recognition and tracking. It can also be extended for segmentation with depth problem [6, 122, 210], using curvature information to recover shapes for occluded objects or establish their ordering relation in a specific scene based on only one single image. Researches [181–184] achieve object detection, recognition and motion estimation in complex background scenarios. However, the existing standard segmentation approaches [40, 41, 142, 171] cannot achieve such illusory contour recovery successfully. They are generally region-based approaches, mainly depending on region features (e.g. gray scale intensity distribution) of an image. Missing edges of internal objects without necessary region features cannot be identified and connected smoothly.

The research work based on a typical partial differential equation (PDE) by Sarti et al., [144] was considered the beginning of variational models for computing illusory contours. They managed to connect many non-existing edges using some fixed points, including some concave and convex points. However, the results of this method are subject to the initialization of the fixed points and the recovered boundaries are straight lines other than smooth curves. Later, Zhu et al., proposed a new model [208] using Eulers elastica [115, 212] based on variational level set method [40, 124, 171], which could recover the illusory contours regardless of whether the missing boundaries are straight lines or curves, even without requiring initialization of fixed points. However, a series of higher order evolution equations are derived in this work, which are extremely difficult to discretize, especially when the fourth-order nonlinear PDEs appeared in their approach. The traditional gradient descent method was utilized to solve these equations and the solution process is extremely slow. In

recent years some related methods have been proposed, e.g., text image deblurring [75, 79] using prior information, and image smoothing [78] via extrema interpolation etc., which inspire us to apply to the variational illusory contour capture model [208] for fast algorithm design.

Other similar works were conducted by Sarti et al. [145] and Jung et al. [82], where they proposed a computational model for segmentation of images with missing boundaries. The approach in [145] was depending on some reference points and their associated surfaces, while authors in [82] designed a low-order model directly based on admissible contours. Both works are edge-based and they have some limitations: the evolutionary curves in these models may not stay on the correct edges if the gray scale values change slightly around edge points, i.e., these approaches were not robust for noisy images.

Motivated by works in [117, 158, 160, 166], we propose an simplified variational level set formulation for illusory contour capture problem. This is achieved by taking the level set functions as signed distance functions. The main difference between the variational level set formulation we proposed and the models in [109, 117, 151, 158, 160, 209, 211] is that our proposed formulation is constrained by some nonlinear Eikonal equations based on definitions of the signed distance functions. Usually, these constrained equations are guaranteed to be satisfied by solving a dynamic HamiltonJacobi equation using the upwind finite difference scheme [154, 155] or the fast sweeping method [204], or the constraints can be dealt with using the penalty function method [93].

The baseline work [208] for capturing illusory contours remains one of the most important and relevant work in the field. Since then, there is few research focusing on fast algorithm in illusory contours capture in terms of level sets. However, there have been many researches [85, 129, 187] aiming to solve the same problem using other techniques, such as point flow or event related potential (ERP) technique. As a classic variational method using 3D implicit equations to represent any closed curve in a plane, there are many advantages of the variational level set approach. 1) Evolution curve can naturally realize topology change along with evolution of the level set function, while the level set function can maintain the effectiveness and one can obtain stable solution; 2) There is no need to track the motion of curves at different moments, since curve evolution can be achieved only through updating level set function in a fixed coordination; 3) It is convenient to calculate certain features (eg. normal vector, curvature etc.) in this approach; and 4) The level set approach can be

easily extended to high dimensional space.

Nonlinear Diffusion Based Image Segmentation

Mumford-Shah model [116] is a milestone in the development process of variational PDEs based image segmentation. Much of the subsequent research progresses are based on it. One important type of methods, the region-based geodesic active contour model, is derived from it. However, the Mumford-Shah model includes two energy terms defined in two-dimensional image space and one-dimensional contour space respectively. There are inevitable difficulties in the course of solving them. A series of approximation approaches are presented which might be roughly classified into three branches. In 1990, Ambrosi and Tortorilli proposed an equivalent model [5] based on Gamma convergence theory by introducing auxiliary variables and approximating the construction of elliptic function. This forms the Gamma-convergence branch for variational image segmentation. The second branch is the variational level set method [166,176,185] based piece-wise constant approximation of the Mumford-Shah model. In 2001, Chan and Vese proposed the two-phase Chan-Vese model [40], which is especially applicable to the circumstance of blurring or discontinuous boundaries, hence it became the most famous model in this branch. The third branch is an extended version of the second one. The typical representative in this branch is the global convex segmentation model, which was transformed from the Chan-Vese model with constraints and the relaxation-based method by Bresson et al. [20, 162].

Nonlinear diffusion technology is a widely used method for image denoising, which has given rise to several significant applications [67,132,207]. During operations, different strategies can be adopted to get better diffusion effects according to the differences between the edge region and the non-edge region. Thus it can adaptively smoothen images regions and enhance edges. In 1990, Perona and Malik creatively introduced the thermal balance equation based anisotropic diffusion model [128]. In their famous research, ϕ was defined as a function. The final result of ϕ obtained through iterative calculation represented the denoised target image. As its corresponding regularization term is $\lambda^2 \ln(1 + |\nabla\phi|^2/\lambda^2)$ (where $\lambda > 0$ is a threshold value), we can know: (a) If $|\nabla\phi| > \lambda$, the pixels are enhanced; (b) If $|\nabla\phi| < \lambda$, the pixels are reduced; (c) If $|\nabla\phi| = \lambda$, the pixels are invariant. In this way, it is effective for both smoothing and edge-preserving in the filtering process. In 1992, Catte et al. [27] extended Perona and Maliks algorithm by applying Gaussian smoothing

to diminish the effect of noise and improve the model adaptability. Subsequently many authors [1, 42, 43, 60, 66, 71, 119, 120, 174] designed various diffusion terms to control the diffusion effects by considering the neighbourhood features based on their respective considerations. It should be noted that the regularization term must be reasonably designed in order to preserve useful details such as edges in the image as much as possible.

Fast optimization schemes

In order to improve the computational efficiency and avoid solving the nonlinear higher-order PDEs, a lot of research work in terms of fast methods [19, 124, 177, 180, 191] have been systematically proposed for energy minimization problems. Among them, the alternating direction method of multiplier (ADMM) played a prominent role due to its attractive properties of simple implementation and broad applicability. For instance, ADMM is widely used to minimize the energy functionals of image denoising [147], inpainting models [191] and some other complicated high-order models [100, 149] fast and effectively. Though a good result could be obtained by the traditional gradient descent flow algorithm, the computation cost utilizing the method of finite difference associated with high-order variables is very expensive. The main objective of ADMM is to solve the constrained optimization problems with multiple variables [61, 188] by minimizing the augmented Lagrangian functional via an alternating direction optimization strategy. The energy functional including high-order variables can be simplified using the introduction of low-order variables which can be computed by the easier iterative algorithm. Thus ADMM not only significantly reduces the order of the variables but also requires fewer steps in its solution process.

The curvature related terms bring extra computational complexity due to the existence of nonlinear higher-order derivatives. This issue also appears in other variational models such as the nontexture image inpainting [191] and image denoising [186] with features (edge, corner, smoothness, contrast, etc.) preservation. Hence, it is essential to take some mathematical optimization techniques, for example, global solution, stability guarantee and calculation acceleration etc., into consideration in the process of algorithm design. In [158], Tai presented a brief review on how to use the ADMM method for solving minimization problems related to curvature. Instead of solving high order nonlinear PDEs, ADMM is applied to transform the energy minimization problem of the proposed model into some separate subproblems, which are then solved alternatively. For each subproblem, many

useful tools such as fast Fourier transform (FFT) [50,114], projection formula [160], analytical soft thresholding equation [164,165,179] and threshold method [32,36] can be applied for further accelerating. In this chapter, fast numerical schemes for optimizing variational energy functionals will be elaborated. Firstly, the standard ADMM is designed for the curvature based image processing model by introducing some auxiliary variables and Lagrange multipliers. Secondly, one type of ADMM developed versions is introduced for the illusory contour capture model, the ADMM-projection (from now on written as ADMM-P) method. It is to simplify the variational level set formulation by replacing curvatures with Laplacians of level set functions. Another salient feature of this method is that the Eikonal equations can be satisfied indirectly by introducing some auxiliary variables and implementing a direct projection [166,190]. The idea of solving the segmentation problem with depth in [166] gives us inspiration for illusory contour reconstruction, and the approach in [190] dealing with Eulers Elastica item provides us some ideas for the proposed ADMM-P approach in this chapter. The proposed ADMM-P is superior compared to algorithms based on traditional gradient descent method (GDM) in terms of computational efficiency. We design two fast algorithms to solve the nonlinear diffusion based general segmentation model: a standard ADMM and a new fast algorithm, a normal vector projection method (NVPM) based on normal vector projection and alternating optimization. To further improve the computing efficiency of NVPM, Nesterovs optimal first-order method [63] is applied.

In this chapter, we propose several models: MTC- ℓ_1 model for image denoising (3.1) and inpainting (3.61), illusory contour capture model (3.35), and nonlinear diffusion based general segmentation model (3.47). Significant performance gains of these proposed models and the advantages of the fast algorithms designed for them will be demonstrated in this chapter via experiments.

The rest of this chapter is organized as follows. From Section 3.2 to Section 3.4, the proposed novel variational models for diverse image tasks and the corresponding algorithm framework are given. In Section 3.5, extensive experiments on natural color images are presented to validate the effectiveness of the proposed models. Summary is given in Section 3.6.

3.2 Proposed Multi-Channel Total Curvature Model for Image Restoration

3.2.1 The Functional of Variational MTC- ℓ_1 Model and its GDM method

Motivated by the development of MTV- ℓ_1 from TV- ℓ_1 for color image de-noising, we attempt to propose a new model based on multi-channel coupled total curvature and ℓ_1 fidelity term, i.e., the MTC- ℓ_1 model for color image restoration with properties of edge, corner, smoothness, contrast and geometry preserving.

Let $\vec{f} = [f_1, f_2, \dots, f_m]^T$ be a color image defined on a domain Ω , and $\vec{u} = [u_1, u_2, \dots, u_m]^T$ be its restored counterpart. Before introducing a formal definition of MTC- ℓ_1 model, we will review the functional of MTV- ℓ_1 model first. Its smoothness term defined in Section 3.2.1 can be expressed as $\int_{\Omega} \sqrt{\sum_{i=1}^m |\nabla u_i|^2} dx$, where $\nabla u_i = [\partial_{x_1} u_i, \partial_{x_2} u_i]^T$, $(x_1, x_2) \in \Omega$. Analog to (2.20) and (2.22), the MTC- ℓ_1 model for color image restoration can be developed as follows based on the coupled total curvature of multi-channels

$$E(\vec{u}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |\kappa_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} |u_i - f_i| dx, \quad (3.1)$$

where $\kappa_i = \nabla \cdot \left(\frac{\nabla u_i}{|\nabla u_i|} \right)$ is curvature which is also presented in (2.22), the first term on the right side is the MTC regularization term, and the second term is the data fidelity term based on ℓ_1 norm. For edge preserving, total curvature possesses similar property with total variation, its value is large on the edge while small on smooth region, an appropriate λ can achieve effective edge keeping. As discussed in (2.22), $|\nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right)|$ contained in the first term can preserve the piece-wise constant normal $\frac{\nabla u}{|\nabla u|}$ of images. Thus, the points corresponding to some discontinuous normals, i.e. the corners in images, can be preserved. AS the similar analysis stated in [209], the evolution equation of (3.1) behaves like the biharmonic heat equation, which can remove the small scale noise then achieve efficient smoothness. Since curvature is in high-order form, which can effectively ease the staircase artifacts. The second term in ℓ_1 form can preserve image contrast as well as remove outliers and irregularities.

By using the standard variational method, the following GDM equations can be obtained

via minimizing the functional (3.1).

$$\frac{\partial u_i}{\partial t} = -\nabla \cdot \left\{ \frac{1}{|\nabla u_i|} \left[\nabla(\Psi'(\kappa)) - \left(\nabla(\Psi'(\kappa)) \cdot \frac{\nabla u_i}{|\nabla u_i|} \right) \frac{\nabla u_i}{|\nabla u_i|} \right] \right\} - \lambda \frac{u_i - f_i}{|u_i - f_i|}, \quad (3.2)$$

where $\Psi(\kappa) = \sqrt{\sum_{i=1}^m |\kappa_i|^2}$ and $\Psi'(\kappa) = \frac{\partial \Psi(\kappa)}{\partial \kappa_i}$. More details on the explicit computation and proofs are given in [38].

Instead of solving (3.1) directly by using the conventional variational method or solving the related GDM equation as mentioned in this section, we transform the original minimization problem into several sub optimization problems by introducing some auxiliary variables and then solve them using an alternating directional optimization strategy.

Our contributions in this section can be summarized as follows:

- A variation model using the multi-channel total curvature based on ℓ_1 norm (MTC- ℓ_1) for color image restoration is proposed, which aims to preserve image contrast, remove anomalies and irregularities (i.e. impulsive noise) from images, eliminate the staircase effect, and preserve edges and corners;
- A coupled regularization terms are used based on curvatures motivated by the concept of MTV model, to achieve edge and corner preserving for color images;
- Instead of solving the nonlinear fourth order partial differential equations directly, we design a fast algorithm based on ADMM by transforming the original energy minimization problem of the proposed MTC- ℓ_1 model into six sub-problems, which are then efficiently solved by FFT, analytical soft threshold formula, and projection methods.

3.2.2 Application of ADMM to MTC- ℓ_1 Model

The ADMM scheme for (3.1) is more complicated than the one for its scalar counterpart (2.22) due to the coupled regularization terms and the associated variables. For clarity, we use the boldface $\vec{\mathbf{x}}$ to denote one composite vector whose components are vectors and the lean type \vec{x} to denote a vector whose components are scalars. Based on the basic idea of ADMM, we now introduce some auxiliary vector $\vec{\mathbf{p}} = [\vec{p}_1^T, \vec{p}_2^T, \dots, \vec{p}_m^T]^T$ with $\vec{p}_i = [p_{i1}, p_{i2}]^T$ and $\vec{\mathbf{n}} = [\vec{n}_1^T, \vec{n}_2^T, \dots, \vec{n}_m^T]^T$ with $\vec{n}_i = [n_{i1}, n_{i2}]^T$ such that $\vec{p}_i = \nabla u_i$ and $\vec{n}_i = \frac{\vec{p}_i}{|\vec{p}_i|}$, which are

solutions of the following constrained minimization problem.

$$\begin{aligned}
\min_{\vec{u}, \vec{p}, \vec{n}} E(\vec{u}, \vec{p}, \vec{n}; \vec{\lambda}_1, \vec{\lambda}_2) &= \int_{\Omega} \sqrt{\sum_{i=1}^m |\nabla \cdot \vec{n}_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} |u_i - f_i| dx \\
&+ \frac{\mu_1}{2} \sum_{i=1}^m \int_{\Omega} \left| |\vec{p}_i| \vec{n}_i - \vec{p}_i \right|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i1} \cdot (|\vec{p}_i| \vec{n}_i - \vec{p}_i) dx \\
&+ \frac{\mu_2}{2} \sum_{i=1}^m \int_{\Omega} |\vec{p}_i - \nabla u_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i2} \cdot (\vec{p}_i - \nabla u_i) dx.
\end{aligned} \tag{3.3}$$

In fact, we aim to use (3.3) to replace the minimization of (3.1) equivalently, where μ_1, μ_2 are predefined penalty parameters, and λ_1, λ_2 are Lagrangian multipliers as $\vec{\lambda}_1 = [\vec{\lambda}_{11}, \vec{\lambda}_{21}, \dots, \vec{\lambda}_{m1}]^T$, $\vec{\lambda}_2 = [\vec{\lambda}_{12}, \vec{\lambda}_{22}, \dots, \vec{\lambda}_{m2}]^T$. However, this model can not be solved directly using the ADMM method due to its large computational costs since there are still some high-order terms in calculation. As discussed by [158, 160], a new auxiliary vector $\vec{m} = [\vec{m}_1, \vec{m}_2, \dots, \vec{m}_m]^T$ with the constraint $\vec{m} \approx \vec{n}$ is introduced, which can be seen as relaxations of \vec{n} in the process of designing a fast and efficient algorithm. In addition, \vec{m} also belongs to the space \mathbb{R}^{2m} , the same as \vec{n} . Here it should be noted that $|\vec{m}_i| \leq 1$ since $\vec{m} \approx \vec{n}$, which is a very important prerequisite requirement. Generally the constraint $|\vec{m}_i| \leq 1$ is guaranteed by solving its steady or dynamic Eikonal equation. In this chapter, a projection method is used with a very explicit form to achieve this. Therefore we can easily obtain $|\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i \geq 0$ from the mathematical point of view. This kind of constraint, which only needs a ℓ_1 -norm for penalization, can be utilized to obtain a simple functional and this is a good way to obtain a relaxed functional. For further simplification, another two auxiliary variables $\vec{q} = [q_1, q_2, \dots, q_m]^T$ and $\vec{v} = [v_1, v_2, \dots, v_m]^T$ are introduced, which are defined in Ω^m such that $q = \vec{\nabla} \cdot \vec{n} = [\nabla \cdot \vec{n}_1, \nabla \cdot \vec{n}_2, \dots, \nabla \cdot \vec{n}_m]^T$ and $\vec{v} = \vec{u} - \vec{f}$. More details will be given in the next subsection.

Finally the corresponding functional can be derived from (3.1) through the ADMM. It can be observed that the obtained functional can successfully avoid dealing with high-order

terms.

$$\begin{aligned}
E(\vec{u}, \vec{q}, \vec{p}, \vec{m}, \vec{n}, \vec{v}; \vec{\lambda}_1, \vec{\lambda}_2, \vec{\lambda}_3, \vec{\lambda}_4, \vec{\lambda}_5) &= \int_{\Omega} \sqrt{\sum_{i=1}^m |q_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} |v_i| dx \quad (3.4) \\
&+ \mu_1 \sum_{i=1}^m \int_{\Omega} \left| |\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i \right|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i1} (|\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i) dx \\
&+ \frac{\mu_2}{2} \sum_{i=1}^m \int_{\Omega} |\vec{p}_i - \nabla u_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i2} \cdot (\vec{p}_i - \nabla u_i) dx \\
&+ \frac{\mu_3}{2} \sum_{i=1}^m \int_{\Omega} |q_i - \nabla \cdot \vec{n}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i3} (q_i - \nabla \cdot \vec{n}_i) dx \\
&+ \frac{\mu_4}{2} \sum_{i=1}^m \int_{\Omega} |\vec{n}_i - \vec{m}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i4} \cdot (\vec{n}_i - \vec{m}_i) dx \\
&+ \frac{\mu_5}{2} \sum_{i=1}^m \int_{\Omega} |v_i - (u_i - f_i)|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i5} [v_i - (u_i - f_i)] dx + \sum_{i=1}^m \delta_{\mathfrak{R}}(\vec{m}_i),
\end{aligned}$$

where, $\vec{\lambda}_1, \vec{\lambda}_2, \vec{\lambda}_3, \vec{\lambda}_4, \vec{\lambda}_5$ are Lagrange multipliers and $\mu_1, \mu_2, \mu_3, \mu_4, \mu_5$ are positive penalty parameters. $\vec{\lambda}_1, \vec{\lambda}_2, \vec{\lambda}_3, \vec{\lambda}_4, \vec{\lambda}_5$ are defined as $\vec{\lambda}_1 = [\lambda_{11}, \lambda_{21}, \dots, \lambda_{m1}]^T$, $\vec{\lambda}_2 = [\vec{\lambda}_{12}, \vec{\lambda}_{22}, \dots, \vec{\lambda}_{m2}]^T$, $\vec{\lambda}_3 = [\lambda_{13}, \lambda_{23}, \dots, \lambda_{m3}]^T$, $\vec{\lambda}_4 = [\vec{\lambda}_{14}, \vec{\lambda}_{24}, \dots, \vec{\lambda}_{m4}]^T$ and $\vec{\lambda}_5 = [\lambda_{15}, \lambda_{25}, \dots, \lambda_{m5}]^T$. \mathfrak{R} is a set defined as $\mathfrak{R} = \{\vec{m}_i \in \mathbf{L}^2(\Omega) \mid |\vec{m}_i| \leq 1 \text{ a.e. in } \Omega\}$ and $\delta_{\mathfrak{R}}(\cdot)$ is a characteristic function on \mathfrak{R} :

$$\delta_{\mathfrak{R}}(\vec{m}_i) = \begin{cases} 0 & \vec{m}_i \in \mathfrak{R} \\ +\infty & \text{otherwise} \end{cases}. \quad (3.5)$$

With these definitions, the constraint $|\vec{m}_i| \leq 1$ has been successfully embedded into the energy functional.

The Minimization for All the Six Subproblems

In order to solve (3.1) efficiently with the ADMM, the unknowns $\vec{u}^0, \vec{q}^0, \vec{p}^0, \vec{m}^0, \vec{n}^0, \vec{v}^0, \vec{\lambda}_1^0, \vec{\lambda}_2^0, \vec{\lambda}_3^0, \vec{\lambda}_4^0, \vec{\lambda}_5^0$ are initialized at $k = 0$. Some minimization problems with respect to one specific variable can be solved while others are temporarily fixed at each step from k to $k+1$ until convergence. This reflects the property of alternating direction optimization approach. Using this idea, the optimization problem (3.1) can be divided into six subproblems of

minimization in the iterative process from the k th to $(k + 1)$ th step:

$$\left\{ \begin{array}{l} \vec{u}^{k+1} = \arg \min \{ \varepsilon_1(\vec{u}) = E(\vec{u}, \vec{q}^k, \vec{p}^k, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \\ \vec{q}^{k+1} = \arg \min \{ \varepsilon_2(\vec{q}) = E(\vec{u}^{k+1}, \vec{q}, \vec{p}^k, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \\ \vec{p}^{k+1} = \arg \min \{ \varepsilon_3(\vec{p}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \\ \vec{m}^{k+1} = \arg \min \{ \varepsilon_4(\vec{m}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \\ \vec{n}^{k+1} = \arg \min \{ \varepsilon_5(\vec{n}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}^{k+1}, \vec{n}, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \\ \vec{v}^{k+1} = \arg \min \{ \varepsilon_6(\vec{v}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}^{k+1}, \vec{n}^{k+1}, \vec{v}; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \} \end{array} \right. \quad (3.6)$$

$$\left\{ \begin{array}{l} \lambda_{i1}^{k+1} = \lambda_{i1}^k + \mu_1(|\vec{p}_i^{k+1}| - \vec{m}_i^{k+1} \cdot \vec{p}_i^{k+1}) \\ \vec{\lambda}_{i2}^{k+1} = \vec{\lambda}_{i2}^k + \mu_2(\vec{p}_i^{k+1} - \nabla u_i^{k+1}) \\ \lambda_{i3}^{k+1} = \lambda_{i3}^k + \mu_3(q_i^{k+1} - \nabla \cdot \vec{n}_i^{k+1}) \\ \vec{\lambda}_{i4}^{k+1} = \vec{\lambda}_{i4}^k + \mu_2(\vec{n}_i^{k+1} - \vec{m}_i^{k+1}) \\ \lambda_{i5}^{k+1} = \lambda_{i5}^k + \mu_5[v_i^{k+1} - (u_i^{k+1} - f_i)] \end{array} \right. \quad (3.7)$$

where

$$\begin{aligned} \varepsilon_1(\vec{u}) &= \frac{\mu_2}{2} \sum_{i=1}^m \int_{\Omega} |\vec{p}_i - \nabla u_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i2} \cdot (\vec{p}_i - \nabla u_i) dx \\ &+ \frac{\mu_5}{2} \sum_{i=1}^m \int_{\Omega} |v_i - (u_i - f_i)|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i5} [v_i - (u_i - f_i)] dx + \sum_{i=1}^m \delta_{\mathfrak{R}}(\vec{m}_i) \end{aligned} \quad (3.8)$$

$$\varepsilon_2(\vec{q}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |q_i|^2} dx + \frac{\mu_3}{2} \sum_{i=1}^m \int_{\Omega} |q_i - \nabla \cdot \vec{n}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i3} (q_i - \nabla \cdot \vec{n}_i) dx \quad (3.9)$$

$$\begin{aligned} \varepsilon_3(\vec{p}) &= \mu_1 \sum_{i=1}^m \int_{\Omega} \left(|\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i \right)^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i1} (|\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i) dx \\ &+ \frac{\mu_2}{2} \sum_{i=1}^m \int_{\Omega} |\vec{p}_i - \nabla u_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i2} \cdot (\vec{p}_i - \nabla u_i) dx \end{aligned} \quad (3.10)$$

$$\begin{aligned}
\varepsilon_4(\vec{m}) &= \mu_1 \sum_{i=1}^m \int_{\Omega} \left| |\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i \right|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i1} (|\vec{p}_i| - \vec{m}_i \cdot \vec{p}_i) dx \\
&+ \frac{\mu_4}{2} \sum_{i=1}^m \int_{\Omega} |\vec{n}_i - \vec{m}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i4} \cdot (\vec{n}_i - \vec{m}_i) dx + \sum_{i=1}^m \delta_{\mathfrak{R}}(\vec{m}_i)
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
\varepsilon_5(\vec{n}) &= \frac{\mu_3}{2} \sum_{i=1}^m \int_{\Omega} |q_i - \nabla \cdot \vec{n}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i3} (q_i - \nabla \cdot \vec{n}_i) dx \\
&+ \frac{\mu_4}{2} \sum_{i=1}^m \int_{\Omega} |\vec{n}_i - \vec{m}_i|^2 dx + \sum_{i=1}^m \int_{\Omega} \vec{\lambda}_{i4} \cdot (\vec{n}_i - \vec{m}_i) dx
\end{aligned} \tag{3.12}$$

$$\varepsilon_6(\vec{v}) = \lambda \sum_{i=1}^m \int_{\Omega} |v_i| dx + \frac{\mu_5}{2} \sum_{i=1}^m \int_{\Omega} |v_i - (u_i - f_i)|^2 dx + \sum_{i=1}^m \int_{\Omega} \lambda_{i5} [v_i - (u_i - f_i)] dx \tag{3.13}$$

In summary, the ADMM can be presented in a pseudo code format as follows. Next we will consider each sub-problem individually.

ADMM for MTC- ℓ_1 model.

Step 1: Initialize the unknown values, $\vec{u}^0, \vec{q}^0, \vec{p}^0, \vec{m}^0, \vec{n}^0, \vec{v}^0, \vec{\lambda}_1^0, \vec{\lambda}_2^0, \vec{\lambda}_3^0, \vec{\lambda}_4^0, \vec{\lambda}_5^0$

Step 2: For $k \geq 1$, solve the following problems alternatively

2.1. Subproblem 1 for \vec{u}^{k+1} :

$$\vec{u}^{k+1} = \arg \min \{ \varepsilon_1(\vec{u}) = E(\vec{u}, \vec{q}^k, \vec{p}^k, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.2. Subproblem 2 for \vec{q}^{k+1} :

$$\vec{q}^{k+1} = \arg \min \{ \varepsilon_2(\vec{q}) = E(\vec{u}^{k+1}, \vec{q}, \vec{p}^k, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.3. Subproblem 3 for \vec{p}^{k+1} :

$$\vec{p}^{k+1} = \arg \min \{ \varepsilon_3(\vec{p}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}, \vec{m}^k, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.4. Subproblem 4 for \vec{m}^{k+1} :

$$\vec{m}^{k+1} = \arg \min \{ \varepsilon_4(\vec{m}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}, \vec{n}^k, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.5. Subproblem 5 for \vec{n}^{k+1} :

$$\vec{n}^{k+1} = \arg \min \{ \varepsilon_5(\vec{n}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}^{k+1}, \vec{n}, \vec{v}^k; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.6. Subproblem 6 for \vec{v}^{k+1} :

$$\vec{v}^{k+1} = \arg \min \{ \varepsilon_6(\vec{v}) = E(\vec{u}^{k+1}, \vec{q}^{k+1}, \vec{p}^{k+1}, \vec{m}^{k+1}, \vec{n}^{k+1}, \vec{v}; \vec{\lambda}_1^k, \vec{\lambda}_2^k, \vec{\lambda}_3^k, \vec{\lambda}_4^k, \vec{\lambda}_5^k) \}$$

2.7. Update Lagrange multipliers using (3.7)

Step 3: The iteration loop will be terminated if the stopping criteria (described in Termination Criteria) are satisfied.

The alternating direction method of multipliers (ADMM) has the capability of reducing the possibility of ill-conditioning [63] and making the numerical computation stable through iterative Lagrangian multiplier during the process of the minimization. In this sec-

tion, (3.8)(3.13) will be solved alternately by different iterative methods and the Lagrange multipliers will be updated in the end.

Subproblem 1: Computing the primal variable \vec{u} . FFT [50, 114] is used for this problem which is known for fast calculating linear equations in order to obtain the minimum values of variables. When \vec{u} is computed, the i th group of auxiliary variables $\vec{q}^k, \vec{p}^k, \vec{m}^k, \vec{n}^k$ and \vec{v}^k are assumed to be fixed. This concept is also applied in the following calculations. The EulerLagrange equation of (3.8) can be obtained as follows:

$$\mu_2 \nabla \cdot (\vec{p}_i^k - \nabla u_i) + \nabla \cdot \vec{\lambda}_{i2}^k - \mu_5 [v_i^k - (u_i - f_i)] - \lambda_{i5}^k = 0. \quad (3.14)$$

As in [114], Eq. (3.14) is a screened Poisson equation for which FFT is a well known solver with very low computational cost for imaging problems. The output of this algorithm is an image obtained by applying the screened Poisson equation to each color channel separately. For this purpose, let us first introduce an instrumental variable $\vec{g} = [g_1, g_2, \dots, g_m]^T$. The calculation procedure is shown as follows:

$$-\mu_2(\partial_{xx}u_i + \partial_{yy}u_i) + \mu_5u_i = g_i, \quad (3.15)$$

where $g_i = -\mu_2 \nabla \cdot \vec{p}_i^k - \nabla \cdot \vec{\lambda}_{i2}^k + \mu_5(v_i^k + f_i) + \lambda_{i5}^k$. Then the identity operator $I\vec{f}(i, j) = \vec{f}(i, j)$ and shifting operators $S_x^\pm \vec{f}(i, j) = \vec{f}(i \pm 1, j)$, $S_y^\pm \vec{f}(i, j) = \vec{f}(i, j \pm 1)$ are used. Eq. (3.15) can be now rewritten as:

$$(-\mu_2(S_x^- - 2I + S_x^+ + S_y^- - 2I + S_y^+) + \mu_5)u_i = g_i. \quad (3.16)$$

By using the properties of discrete Fourier transform, the corresponding relations between the Fourier transforms of the shifting operators and those of original functions $\vec{f}(y_i, y_j)$ can be given as: $FFT S_x^\pm \vec{f}(y_i, y_j) = e^{\pm i Z_i} FFT \vec{f}(y_i, y_j)$ and $FFT S_y^\pm \vec{f}(y_i, y_j) = e^{\pm i Z_j} FFT \vec{f}(y_i, y_j)$. y_i and y_j are discrete frequencies. $Z_i = \frac{2\pi}{N_1} y_i$, $y_i = 1, 2, \dots, N_1$ and $Z_j = \frac{2\pi}{N_2} y_j$, $y_j = 1, 2, \dots, N_2$. An algebraic equations can be obtained as:

$$(-2\mu_2(\cos Z_i + \cos Z_j - 2) + \mu_5)FFT(u_i) = FFT(g_i). \quad (3.17)$$

It can be proved that the fast Fourier transform can be directly used in Ω . The discrete

inverse Fourier transform IFFT will be used to update u^k as follows.

$$u_i^{k+1} = \Re \left(IFFT \left(\frac{FFT(g_i)}{(-2\mu_2(\cos Z_i + \cos Z_j - 2) + \mu_5)} \right) \right), \quad (3.18)$$

where $\Re(\cdot)$ represents the real part of a complex number.

Subproblem 2: Minimization with respect to \vec{q} . The soft threshold formula proposed in [164, 165, 179] is used here to calculate the variable \vec{q} . It is one of the most classical algorithms, which has been used widely to obtain the minimum values of the variables. Here the variables \vec{u}^{k+1} , \vec{p}^k , \vec{m}^k, \vec{n}^k and \vec{v}^k are fixed and we aim to calculate q . As the EulerLagrange equation of (3.9) is given:

$$\frac{q_i}{\sqrt{\sum_{i=1}^m |q_i|^2}} + \mu_3 \left(q_i - \nabla \cdot \vec{n}_i^k + \frac{\lambda_{i3}^k}{\mu_3} \right) = 0. \quad (3.19)$$

The calculation result can be obtained via the generalized soft threshold formula as:

$$q_i^{k+1} = \max \left(\sqrt{\sum_{i=1}^m (\nabla \cdot \vec{n}_i^k - \frac{\lambda_{i3}^k}{\mu_3})^2} - \frac{1}{\mu_3}, 0 \right) \frac{\nabla \cdot \vec{n}_i^k - \frac{\lambda_{i3}^k}{\mu_3}}{\sqrt{\sum_{i=1}^m (\nabla \cdot \vec{n}_i^k - \frac{\lambda_{i3}^k}{\mu_3})^2}}. \quad (3.20)$$

Subproblem 3: The minimization with respect to \vec{p} . The variables of \vec{u}^{k+1} , \vec{q}^{k+1} , \vec{m}^k, \vec{n}^k and \vec{v}^k are fixed here. The EulerLagrange equation of (3.10) is:

$$(\mu_1 + \lambda_{i1}^k) \frac{\vec{p}_i}{|\vec{p}_i|} + \mu_2 \left(\vec{p}_i - \nabla u_i^{k+1} + \frac{\vec{\lambda}_{i2}^k}{\mu_2} - \frac{(\mu_1 + \lambda_{i1}^k) \vec{m}_i^k}{\mu_2} \right) = 0, \quad (3.21)$$

\vec{p}^{k+1} can be obtained in the similar way as (3.20). The calculation results are given below:

$$\vec{p}_i^{k+1} = \max \left(\left| \nabla u_i^{k+1} + \frac{(\mu_1 + \lambda_{i1}^k) \vec{m}_i^k - \vec{\lambda}_{i2}^k}{\mu_2} \right| - \frac{(\mu_1 + \lambda_{i1}^k)}{\mu_2}, 0 \right) \frac{\nabla u_i^{k+1} + \frac{(\mu_1 + \lambda_{i1}^k) \vec{m}_i^k - \vec{\lambda}_{i2}^k}{\mu_2}}{\left| \nabla u_i^{k+1} + \frac{(\mu_1 + \lambda_{i1}^k) \vec{m}_i^k - \vec{\lambda}_{i2}^k}{\mu_2} \right|}, \quad (3.22)$$

Subproblem 4: The minimization with respect to \vec{m} . According to (3.11), when \vec{u}^{k+1} , \vec{q}^{k+1} , \vec{p}^{k+1} , \vec{n}^k and \vec{v}^k are fixed, it is very easy to obtain the expansion of its EulerLagrange

equation.

$$-(\mu_1 + \lambda_{i1}^k) \bar{p}_i^{k+1} - \bar{\lambda}_{i4}^k - \mu_4 (\bar{n}_i^k - \bar{m}_i) + \delta_{\mathfrak{R}}(\bar{m}_i) = 0. \quad (3.23)$$

According to [160], we can obtain the closed-form solution of (3.23) through the following equations. A simple projection method is used for guaranteeing the constraint of \bar{m}_i^{k+1} .

$$\begin{cases} \bar{m}_i^{k+1} = \text{proj}_{\mathfrak{R}}(\bar{m}_i^{k+1}) = \begin{cases} \bar{m}_i^{k+1} & |\bar{m}_i^{k+1}| \leq 1 \\ \frac{\bar{m}_i^{k+1}}{|\bar{m}_i^{k+1}|} & \text{otherwise} \end{cases} = \frac{\bar{m}_i^{k+1}}{\max(|\bar{m}_i^{k+1}|, 1)} \\ \text{where } \bar{m}_i^{k+1} = \bar{n}_i^k + \frac{(\mu_1 + \lambda_{i1}^k) \bar{p}_i^{k+1} + \bar{\lambda}_{i4}^k}{\mu_4} \end{cases} \quad (3.24)$$

Subproblem 5: The minimization with respect to \bar{n} . When dealing with the iterations for \bar{n} , fast Fourier transform algorithm is used with the assumption that \bar{u}^{k+1} , \bar{q}^{k+1} , \bar{p}^{k+1} , \bar{m}^{k+1} and \bar{v}^{k+1} are fixed. The EulerLagrange equation of (3.12) is then obtained as follows:

$$\nabla(q_i^{k+1} - \nabla \cdot \bar{n}_i) + \frac{\nabla \lambda_{i3}^k}{\mu_3} + \frac{\mu_4}{\mu_3} (\bar{n}_i - \bar{m}_i^{k+1}) + \frac{\bar{\lambda}_{i4}^k}{\mu_3} = 0. \quad (3.25)$$

According to the definition of (3.14) and (3.15), the coupled Eqs. (3.25) can be expressed as two discrete forms:

$$\begin{cases} -(\partial_{xx} n_{i1} + \partial_{xy} n_{i2}) + \frac{\mu_4}{\mu_3} n_{i1} = -\partial_x q_i^{k+1} - \frac{\partial_x \lambda_{i3}^k}{\mu_3} + \frac{\mu_4}{\mu_3} m_{i1} - \frac{\lambda_{i41}^k}{\mu_3} \\ -(\partial_{yx} n_{i1} + \partial_{yy} n_{i2}) + \frac{\mu_4}{\mu_3} n_{i2} = -\partial_y q_i^{k+1} - \frac{\partial_y \lambda_{i3}^k}{\mu_3} + \frac{\mu_4}{\mu_3} m_{i2} - \frac{\lambda_{i42}^k}{\mu_3} \end{cases}. \quad (3.26)$$

By using the shifting operators mentioned in Subproblem 1, (3.26) can be written as:

$$\begin{cases} -((S_x^- - 2I + S_x^+) n_{i1} + (S_x^+ - S_x^+ S_y^- - I + S_y^-) n_{i2}) + \frac{\mu_4}{\mu_3} n_{i1} = h_{i1} \\ -((S_y^+ - S_x^- S_y^+ - I + S_x^-) n_{i1} + (S_y^- - 2I + S_y^+) n_{i2}) + \frac{\mu_4}{\mu_3} n_{i2} = h_{i2} \end{cases}, \quad (3.27)$$

where $h_{i1} = -\partial_x q_i^{k+1} - \frac{\partial_x \lambda_{i3}^k}{\mu_3} + \frac{\mu_4}{\mu_3} m_{i1} - \frac{\lambda_{i41}^k}{\mu_3}$ and $h_{i2} = -\partial_y q_i^{k+1} - \frac{\partial_y \lambda_{i3}^k}{\mu_3} + \frac{\mu_4}{\mu_3} m_{i2} - \frac{\lambda_{i42}^k}{\mu_3}$. By applying the discrete Fourier transform to both sides of (3.27), we will have the following

system of linear equation.

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} FFT(n_{i1}) \\ FFT(n_{i2}) \end{pmatrix} = \begin{pmatrix} FFT(h_{i1}) \\ FFT(h_{i2}) \end{pmatrix}, \quad (3.28)$$

with the coefficients :

$$\begin{cases} a_{11} = \frac{\mu_4}{\mu_3} - 2(\cos Z_i - 1) \\ a_{12} = -(1 - \cos Z_j + \sqrt{-1} \sin Z_j)(-1 + \cos Z_i + \sqrt{-1} \sin Z_i) \\ a_{21} = -(1 - \cos Z_i + \sqrt{-1} \sin Z_i)(-1 + \cos Z_j + \sqrt{-1} \sin Z_j) \\ a_{22} = \frac{\mu_4}{\mu_3} - 2(\cos Z_i - 1) \end{cases}.$$

The determinant of the coefficient matrix is $D = \left(\frac{\mu_4}{\mu_3}\right)^2 - \frac{2\mu_4}{\mu_3}(\cos Z_i + \cos Z_j - 2)$, which proves to be always positive for all discrete frequencies if $\frac{\mu_4}{\mu_3} > 0$. After the linear equations are solved for each frequency, we can use the discrete inverse Fourier transform to obtain \vec{n} :

$$\begin{cases} n_{i1} = \Re\left(IFTT\left(\frac{a_{22}FFT(h_{i1}) - a_{12}FFT(h_{i2})}{D}\right)\right) \\ n_{i2} = \Re\left(IFTT\left(\frac{-a_{21}FFT(h_{i1}) + a_{11}FFT(h_{i2})}{D}\right)\right) \end{cases}. \quad (3.29)$$

Subproblem 6: The minimization with respect to \vec{v} . Now it is time to consider the last set of variables v in Eq. (3.13). Let the five calculated variables \vec{u}^{k+1} , \vec{q}^{k+1} , \vec{p}^{k+1} , \vec{m}^{k+1} and \vec{n}^{k+1} be fixed, then the EulerLagrange equation of (3.13) can be obtained as:

$$\lambda \frac{v_i}{|v_i|} + \mu_5 \left(v_i - u_i^{k+1} + f_i + \frac{\lambda_{i5}^k}{\mu_5} \right) = 0. \quad (3.30)$$

It is obvious that the exact soft threshold formula, which has been used to obtain \vec{q}^{k+1} and \vec{p}^{k+1} , can be used to solve (3.30). \vec{n}^{k+1} is then obtained based on the solution for \vec{q}^{k+1} .

$$v_i^{k+1} = \max \left(\left| u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5} \right| - \frac{\lambda}{\mu_5}, 0 \right) \frac{u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5}}{\left| u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5} \right|}. \quad (3.31)$$

Update of Lagrange multipliers. At the end of the implementation, the Lagrange multipliers $\vec{\lambda}_1^k$, $\vec{\lambda}_2^k$, $\vec{\lambda}_3^k$, $\vec{\lambda}_4^k$ and $\vec{\lambda}_5^k$ should be updated. In fact, they can be directly calculated through (3.7).

Termination Criteria

As described in [158,166], the iterations in the proposed algorithm need to be terminated when the following criteria are satisfied:

- i. the constraint errors should be monitored in iterations:

$$(R_1^k \quad R_2^k \quad R_3^k \quad R_4^k \quad R_5^k) = \left(\frac{\|\vec{R}_1^k\|_{\ell_1}}{\|\Omega\|_{\ell_1}} \quad \frac{\|\vec{R}_2^k\|_{\ell_1}}{\|\Omega\|_{\ell_1}} \quad \frac{\|\vec{R}_3^k\|_{\ell_1}}{\|\Omega\|_{\ell_1}} \quad \frac{\|\vec{R}_4^k\|_{\ell_1}}{\|\Omega\|_{\ell_1}} \quad \frac{\|\vec{R}_5^k\|_{\ell_1}}{\|\Omega\|_{\ell_1}} \right) \quad (3.32)$$

with

$$\begin{aligned} \vec{R}_1^k &= [R_{11}^k \dots R_{m1}^k]^r = [|\vec{p}_1^k| - \vec{m}_1^k \cdot \vec{p}_1^k \dots |\vec{p}_m^k| - \vec{m}_m^k \cdot \vec{p}_m^k]^T \\ \vec{R}_2^k &= [R_{12}^k \dots R_{m2}^k]^r = [\vec{p}_1^k - \nabla u_1^k \dots \vec{p}_m^k - \nabla u_m^k]^T \\ \vec{R}_3^k &= [R_{13}^k \dots R_{m3}^k]^r = [q_1^k - \nabla \cdot \vec{n}_1^k \dots q_m^k - \nabla \cdot \vec{n}_m^k]^T \\ \vec{R}_4^k &= [R_{14}^k \dots R_{m4}^k]^r = [\vec{n}_1^k - \vec{m}_1^k \dots \vec{n}_m^k - \vec{m}_m^k]^T \\ \vec{R}_5^k &= [R_{15}^k \dots R_{m5}^k]^r = [v_1^k - (u_1^k - f_1) \dots v_m^k - (u_m^k - f_m)]^T \end{aligned}$$

where $\|\cdot\|_{\ell_1}$ denotes the ℓ_1 norm on image domain Ω and $\|\Omega\|_{\ell_1}$ is the area of domain. All components in (63) are calculated point by point. If $R_i^k < \varepsilon$ (ε is a small enough parameter) for $i = 1, 2, \dots, 5$, the iteration of outer repeat k should be stopped. These numerical indicators are also used to determinate the values of μ_i ($i = 1, 2, \dots, 5$), which can be used as the basis of penalty parameter adjustment.

- ii. During the iteration, the relative errors of Lagrange multipliers and the solution \vec{u}^k should be observed. They should reduce to a sufficiently small level:

$$L_i^k = \frac{\|\vec{\lambda}_i^k - \vec{\lambda}_i^{k-1}\|_{\ell_1}}{\|\vec{\lambda}_i^{k-1}\|_{\ell_1}} \quad (i = 1, 2, \dots, 5) \quad (3.33)$$

$$\frac{\|u_i^k - u_i^{k-1}\|_{\ell_1}}{\|u_i^{k-1}\|_{\ell_1}}$$

- iii. The relative energy error can also be chosen as the stopping criterion

$$\frac{\|E^k - E^{k-1}\|_{\ell_1}}{\|E^{k-1}\|_{\ell_1}} \leq \varepsilon \quad (3.34)$$

where E^k is the energy value of (3.4). The computation stops automatically when $\frac{\|E^k - E^{k-1}\|_{\ell_1}}{\|E^{k-1}\|_{\ell_1}}$ is less than a predefined tolerance, which implies the energy approaches its steady state.

3.3 Proposed Simplified Variational Level Set Formulation for Illusory Contour Reconstruction

3.3.1 A New Formulation for Illusory Contour Capture and its GDM method

The original problem (2.23) can be formulated in an equivalently reduced version by taking the signed distance functions as level set functions. If the level set functions are treated as the signed distance functions with $|\nabla\varphi| = 1$ during computation, all the curvatures in (2.23) can be replaced by Laplacians of level set functions. In this case, that is, $\kappa_i = \nabla \cdot \left(\frac{\nabla\varphi_i}{|\nabla\varphi_i|} \right) = \nabla \cdot (\nabla\varphi) = \Delta\varphi$. Therefore, the curvature will be reduced to $\kappa = \Delta\varphi$. According to this simplified form, the term $\kappa(d)$ will be reduced to Δd and it satisfies $\kappa^+(d) = \max(\kappa(d), 0) = \max(\Delta d, 0) = (\Delta d)^+$. With the constraint $|\nabla\varphi| = 1$ in (3.35) below, we can directly use the value $|\nabla\varphi| = 1$ to replace $|\nabla\varphi|$ in Functional (2.23). Based on this observation, the illusory contour capture functional (3.35) can be rewritten as the following simplified version

$$\begin{aligned}
 E(\varphi) &= \int_{\Omega} (1 + \mu C(d)(\Delta d)^+) |d| \delta(\varphi) |\nabla\varphi| dx + \lambda \int_{\Omega} H(d) H(\varphi) dx \\
 &\quad + \int_{\Omega} (a + b(\Delta\varphi)^2) \delta(\varphi) dx. \tag{3.35} \\
 &\quad \text{s.t.} \quad |\nabla\varphi| = 1
 \end{aligned}$$

It should be noted that Eq. (3.35) are equivalent to the original problem (2.23). A novel model with reduced order terms is thus obtained with the constraint $|\nabla\varphi| = 1$. In this case, the nonlinear fourth-order derivatives disappear and the re-initialization procedure of the level set function can be avoided by adding the constraint instead. The computational cost is hence reduced to some extent based on this new model.

In order to solve (2.23), Zhu [208] used the standard variational method to obtain the following GDM equation for the final function φ :

$$\begin{aligned}
 \frac{\partial\varphi}{\partial t} &= \nabla[(1 + \mu C(d)\kappa^+(d))|d|] \cdot \nabla\varphi - \lambda H(d)\delta(\varphi) \tag{3.36} \\
 &\quad + [(1 + \mu C(d)\kappa^+(d))|d|] |\nabla\varphi| \nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|} \right) \\
 &\quad + |\nabla\varphi| \nabla \cdot \left[\frac{\nabla\varphi}{|\nabla\varphi|} \Psi(\kappa) - \frac{1}{|\nabla\varphi|} \left(\nabla(\Psi'(\kappa)|\nabla\varphi|) - \left(\frac{\nabla\varphi}{|\nabla\varphi|} \cdot \nabla(\Psi'(\kappa)|\nabla\varphi|) \right) \frac{\nabla\varphi}{|\nabla\varphi|} \right) \right],
 \end{aligned}$$

where $\Psi(\kappa) = a + b\kappa^2$.

There is a fourth order derivative in equations (3.2) and (3.36), which brings much difficulty in solving them. In fact, there are three main computational disadvantages related to the GDM equations (3.2) and (3.36) as listed below.

- (1) Equation (3.2) and (3.36) are in the form of a standard Hamilton-Jacobi equation, which can be solved iteratively using some explicit upwind-alike schemes. However, the convergence of such iteration depends heavily on discretization of the nonlinear fourth-order derivatives in the GDM equations above, which is proven to be tedious and prone to errors;
- (2) The property of the signed distance function was not utilized in the GDM algorithm proposed in [208]. Specifically, the curvature item in (2.23) can be simplified via the property of signed distance function (e.g. $|\nabla\varphi| = 1$) and then the model can be simplified. After model simplification, the high-order derivatives mentioned above can be avoided and the computational process can be simplified significantly;
- (3) Another critical issue in (3.36) is the inevitable re-initialization procedure of the level set function φ during the evolution process. It requires employing (2.25) regularly for φ with $u = \varphi(x, 0)$ in order to keep φ close to the signed distance function. This is a critical issue in GDM approach and it has been discussed in many papers [93, 166]. Such a process is removed in our proposed ADMM-P approach designed for solving (3.35) numerically.

Our contributions in this section can be summarized as follows:

- A simplified variational level set formulation is proposed by taking advantage of the property of signed distance functions;
- A fast ADMM method for optimizing the transformed model is developed. The main difference between the models in [109, 117, 158, 211] and our proposed model is that our proposed formulation is constrained by some nonlinear Eikonal equations based on definitions of the signed distance functions.

3.3.2 ADMM-P Algorithm for Illusory Contour Capture model

To improve the efficiency further, some fast algorithms can be designed and directly applied to the equivalently simplified version (3.35). As many models containing the Eulers elastica term are able to recover the noisy images with excellent performance, we also apply our model for noisy images with minor modification. Experiments confirm that the proposed model can achieve better performance on noisy images due to the simplified computation procedure. We will follow the main ideas of ADMM combining with a proposed projection method (i.e. ADMM-P algorithm) to solve the proposed model (3.35). Such operations can remove the need of re-initialization for the level set function.

Firstly, some auxiliary variables are introduced, for example, $\vec{w} = [w_1, w_2]^T$ and v with property $\vec{w} \approx \nabla\varphi$, $v \approx \nabla \cdot \vec{w}$ and the Lagrangian multipliers $\vec{\lambda}_1, \lambda_2$. The constraint $|\nabla\varphi| = 1$ can be replaced by $|\vec{w}| = 1$, and this substitution would avoid the traditional re-initialization process and would force $|\nabla\varphi| = 1$ by imposing the compulsory constraint for \vec{w} . $|\vec{w}| = 1$ can then be guaranteed automatically by implementing a projection. Based on this observation, (3.35) can be transformed into the following augmented Lagrangian functional with some explicit constraints:

$$\begin{aligned}
& E(\varphi, \vec{w}, v; \vec{\lambda}_1, \lambda_2) \tag{3.37} \\
& = \int_{\Omega} [(1 + \mu C(d)(\Delta d)^+) |d| \delta(\varphi) + \lambda H(d) H(\varphi)] dx + \int_{\Omega} [a + b(v)^2] \delta(\varphi) dx \\
& + \int_{\Omega} \vec{\lambda}_1 \cdot (\vec{w} - \nabla\varphi) dx + \frac{\mu_1}{2} \int_{\Omega} (\vec{w} - \nabla\varphi)^2 dx \\
& + \int_{\Omega} \lambda_2 (v - \nabla \cdot \vec{w}) dx + \frac{\mu_2}{2} \int_{\Omega} (v - \nabla \cdot \vec{w})^2 dx \\
& s.t. \quad |\vec{w}| = 1
\end{aligned}$$

where μ_1, μ_2 are positive penalty parameters and $\vec{\lambda}_1 = [\lambda_{11}, \lambda_{12}]$. The approximations $\vec{w} \approx \nabla\varphi$ and $v \approx \nabla \cdot \vec{w}$ can be achieved by the maximization with respect to $\vec{\lambda}_1$ and λ_2 in the energy functional (3.37). Meanwhile, the constraint $|\nabla\varphi| = 1$ is replaced by $|\vec{w}| = 1$, which can be guaranteed using the projection directly. It is noteworthy that this constraint can also be guaranteed using the penalty function method, in which larger computational expense would be inevitable. Next we propose a new approach to solve this functional.

Similarly, the proposed ADMM-P method can be implemented in finite steps with stopping criteria. In each step, a sub-problem of minimization is carried out with respect to one

variable while keeping other variables constant temporarily. For the formulations of (3.37), the unknown values φ^0 , \bar{w}^0 , v^0 , $\bar{\lambda}_1^0$, λ_2^0 are initialized, then a procedure of optimization is executed step by step. In each step, the calculation of (3.37) can be divided into the following sub-problems

$$\begin{cases} \varphi^{k+1} = \arg \min\{\varepsilon_1(\varphi) = E(\varphi, \bar{w}^k, v^k; \bar{\lambda}_1^k, \lambda_2^k)\} \\ \bar{w}^{k+1} = \arg \min\{\varepsilon_2(\bar{w}) = E(\varphi^{k+1}, \bar{w}, v^k; \bar{\lambda}_1^k, \lambda_2^k)\} \\ v^{k+1} = \arg \min\{\varepsilon_3(v) = E(\varphi^{k+1}, \bar{w}^{k+1}, v; \bar{\lambda}_1^k, \lambda_2^k)\} \end{cases} \quad (3.38)$$

$$\begin{cases} \bar{\lambda}_1^{k+1} = \bar{\lambda}_1^k + \mu_1(\bar{w}^{k+1} - \nabla\varphi^{k+1}) \\ \lambda_2^{k+1} = \lambda_2^k + \mu_2(v^{k+1} - \nabla \cdot \bar{w}^{k+1}) \end{cases} \quad (3.39)$$

where (3.39) are used for updating the Lagrange multipliers. The functionals of the above mentioned subproblems are given respectively below:

$$\begin{aligned} \varepsilon_1(\varphi) &= \int_{\Omega} [(1 + \mu C(d)(\Delta d)^+) |d| \delta(\varphi) + \lambda H(d) H(\varphi)] dx \\ &+ \int_{\Omega} [a + b(v)^2] \delta(\varphi) dx + \int_{\Omega} \bar{\lambda}_1 \cdot (\bar{w} - \nabla\varphi) dx + \frac{\mu_1}{2} \int_{\Omega} (\bar{w} - \nabla\varphi)^2 dx \end{aligned} \quad (3.40)$$

$$\begin{aligned} \varepsilon_2(\bar{w}) &= \int_{\Omega} \bar{\lambda}_1 \cdot (\bar{w} - \nabla\varphi) dx + \frac{\mu_1}{2} \int_{\Omega} (\bar{w} - \nabla\varphi)^2 dx \\ &+ \int_{\Omega} \lambda_2 (v - \nabla \cdot \bar{w}) dx + \frac{\mu_2}{2} \int_{\Omega} (v - \nabla \cdot \bar{w})^2 dx \end{aligned} \quad (3.41)$$

$$\varepsilon_3(v) = \int_{\Omega} [a + b(v)^2] \delta(\varphi) dx + \int_{\Omega} \lambda_2 (v - \nabla \cdot \bar{w}) dx + \frac{\mu_2}{2} \int_{\Omega} (v - \nabla \cdot \bar{w})^2 dx \quad (3.42)$$

In summary, we can present the proposed ADMM-P approach in a pseudo code format as follows.

Minimization of Each Subproblem

Subproblem 1: Calculation of the level set function φ . For the minimization of subproblems (3.40) with respect to the level set function φ , the corresponding Euler-Lagrange

ADMM-P for the illusory contours capture functional.

Step 1: Initialization: $\varphi^0, \vec{w}^0, v^0, \vec{\lambda}_1^0, \lambda_2^0$

Step 2: For $k \geq 1$, solve the following problems alternatively

2.1. Subproblem 1: update φ^{k+1} via

$$\varphi^{k+1} = \arg \min \{ \varepsilon_1(\varphi) = E(\varphi, \vec{w}^k, v^k; \vec{\lambda}_1^k, \lambda_2^k) \}$$

2.2. Subproblem 2: update \vec{w}^{k+1} via

$$\vec{w}^{k+1} = \arg \min \{ \varepsilon_2(\vec{w}) = E(\varphi^{k+1}, \vec{w}, v^k; \vec{\lambda}_1^k, \lambda_2^k) \}$$

2.3. Subproblem 3: update v^{k+1} via

$$v^{k+1} = \arg \min \{ \varepsilon_3(v) = E(\varphi^{k+1}, \vec{w}^{k+1}, v; \vec{\lambda}_1^k, \lambda_2^k) \}$$

2.4. Subproblem 4: Update Lagrange multipliers via (3.39)

Step 3: Terminate the above iterations if the stopping criteria (described in Termination Criteria) are satisfied.

equations are given by

$$\left\{ \begin{array}{l} (1 + \mu C(d)(\Delta d)^+) |d| \delta'(\varphi) + \lambda H(d) \delta(\varphi) + [a + b(v^k)^2] \delta'(\varphi) \\ + \nabla \cdot \vec{\lambda}_1^k + \mu_1 \nabla \cdot (\vec{w}^k - \nabla \varphi) = 0 \quad \text{in } \Omega \\ (\vec{\lambda}_1^k + \mu_1 (\vec{w}^k - \nabla \varphi)) \vec{n} = 0 \quad \text{on } \partial \Omega \end{array} \right. , \quad (3.43)$$

where \vec{w}^k and v^k are fixed temporarily when φ^{k+1} is calculated by using the semi-implicit difference scheme and Gauss-Seidel iterative method. According to [107], $|\nabla \varphi|$ can be used to replace $\delta(\varphi)$, which accelerates the evolution process, and can be directly normalized via $|\nabla \varphi| = 1$. From Subproblem 2 to 3, similar calculation of ADMM for MTC- ℓ_1 model can be implemented here. We just present the obtained results, the whole solving process can be found in [164].

Subproblem 2: Minimization of the auxiliary variable \vec{w} . The Euler-Lagrange equation with respect to \vec{w} can be derived as below from (3.41) via calculus of variation when the variables φ^{k+1} and v^k are kept constant temporarily,

$$\left\{ \begin{array}{l} \vec{\lambda}_1^k + \mu_1 (\vec{w} - \nabla \varphi^{k+1}) + \nabla \lambda_2^k + \mu_2 \nabla (v^k - \nabla \cdot \vec{w}) = 0 \\ \text{s.t. } |\vec{w}| = 1 \end{array} \right. . \quad (3.44)$$

Then we can obtain:

$$\left\{ \begin{array}{l} \tilde{w}_1^{k+1} = \Re \left(IFFT \left(\frac{a_{22} FFT(g_1) - a_{12} FFT(g_2)}{D} \right) \right) \\ \tilde{w}_2^{k+1} = \Re \left(IFFT \left(\frac{-a_{21} FFT(g_1) + a_{11} FFT(g_2)}{D} \right) \right) \end{array} \right. . \quad (3.45)$$

The determinant of the coefficient matrix is $D = \mu_1^2 - 2\mu_1\mu_2(\cos Z_i + \cos Z_j - 2)$. This value is always positive for all discrete frequencies if $\mu_1 > 0$. Finally, an important projection technique is used to guarantee the constraint $|\vec{w}| = 1$ to be satisfied if we define

$$\vec{w}^{k+1} = \frac{\vec{w}^{k+1}}{|\vec{w}^{k+1}|}. \quad (3.46)$$

The constraint $|\nabla\varphi| = 1$ can be forced by imposing the compulsory constraint for \vec{w} , hence the traditional re-initialization process can be avoided through this projection.

Subproblem 3: Minimization of the auxiliary variable v . For the minimization of (3.42), derive the Euler-Lagrange equation with respect to v while fixing the variables φ^{k+1} , \vec{w}^{k+1} temporarily.

Updating the Lagrange multipliers. The Lagrange multipliers $\vec{\lambda}_1^k$ and λ_2^k need to be updated via (3.39) after all sub-minimization problems have achieved their minimum.

Termination Criteria

For The stopping criteria for each sub-problem, please refer to the termination criteria section given in Section 3.2.2 or [164].

Some specific methods are used to tune parameters in the implementation process of the proposed approach. The two parameters in Functional (3.37), a and b , control the length and curvature of the segmentation boundary. The ratio between a and b is related to the connectivity and smoothness of the level lines. As discussed in [151, 158, 160], the connection of disconnected level lines and smoothness of level lines can be guaranteed by a large parameter b . In addition, another two parameters: μ_1 and μ_2 associated with Lagrange multipliers in the augmented Lagrange energy functional (3.37) also should be tuned, as discussed in Section 3.2.2.

3.4 Proposed Nonlinear diffusion based models for Image Segmentation

3.4.1 Nonlinear Diffusion based Image Segmentation

Considering the advantages of using nonlinear diffusion as the regularization and the good effects of active contour model mentioned above, we propose a novel model which combines

these two technologies to get better effects for image segmentation. This variational model for image segmentation is based on active contours [40], nonlinear diffusion (Table 2.1) and level sets [176,210]. It includes a Chan-Vese model-based data fitting term and a regularized term that uses the potential functions of nonlinear diffusion. The former term can segment the image by region partition instead of having to rely on the edge information. The latter term is capable of automatically preserving image edges as well as smoothing noisy regions. Thus the variational formulation of our nonlinear segmentation model can be written as:

$$E(u, \phi) = \int_{\Omega} r(u_1, u_2)\phi dx + \gamma \int_{\Omega} \varphi(|\nabla\phi|)dx. \quad (3.47)$$

Our contributions in this section can be summarized as follows:

- A general variational model for image segmentation based on nonlinear diffusion technology and Chan-Vese model is proposed to obtain a higher quality of segmentation results;
- ADMM is applied to transform the energy minimization problem of proposed model into three subproblems. Moreover, we creatively propose a new fast algorithm, NVPM, based on normal vector projection and alternating optimization method to solve our model. To further improve the computing efficiency, Nesterovs optimal first-order method [63] is applied into NVPM.

3.4.2 Design of ADMM and NVPM for the Nonlinear Diffusion Based Image Segmentation.

I. ADMM for the Nonlinear Diffusion Based Image Segmentation

The implementation of ADMM for (3.47) is discussed briefly here as it has been already described in detail previously. Firstly, an auxiliary vector $\vec{w} = [w_1, w_2]^T$ is introduced such that $\vec{w} \approx \nabla\phi = [\partial_x\phi, \partial_y\phi]^T$ to transform the functional (3.47) into a constrained problem as follows:

$$\begin{aligned} E(u, \phi, \vec{w}; \vec{\lambda}) &= \int_{\Omega} r(u_1, u_2)\phi dx + \gamma \int_{\Omega} \varphi(|\vec{w}|)dx \\ &+ \frac{\mu}{2} \int_{\Omega} |\vec{w} - \nabla\phi|^2 dx + \int_{\Omega} \vec{\lambda} \cdot (\vec{w} - \nabla\phi) dx, \\ \text{s.t. } &\vec{w} = \nabla\phi \end{aligned} \quad (3.48)$$

The implementation of ADMM is summarized in pseudo code as follows.

ADMM for the nonlinear diffusion based image segmentation

1. Initialization: $u^0 = 0, \phi^0 \in \{0, 1\}, \bar{w}^0 = 0$ and $\bar{\lambda}^0 = 0$.
2. For $k = 0, 1, \dots$ solving the following problems alternatively:
 - 2.1. Subproblem 1 about u^{k+1} :

$$u^{k+1} = \operatorname{argmin}\{\varepsilon_1(u) = E(u, \phi^k, \bar{w}^k; \bar{\lambda}^k)\}.$$
 - 2.2. Subproblem 2 about ϕ^{k+1} :

$$\begin{aligned} \phi^{k+1} &= \operatorname{argmin}\{\varepsilon_2(\phi) = E(u^{k+1}, \phi, \bar{w}^k; \bar{\lambda}^k)\}. \\ \phi^{k+1} &= \prod_{\phi}(\phi^{k+1}). \end{aligned}$$
 - 2.3. Subproblem 3 about \bar{w}^{k+1} :

$$\bar{w}^{k+1} = \operatorname{argmin}\{\varepsilon_3(\phi) = E(u^{k+1}, \phi^{k+1}, \bar{w}; \bar{\lambda}^k)\}.$$
 - 2.4. Update Lagrange multiplier $\bar{\lambda}^{k+1}$:

$$\bar{\lambda}^{k+1} = \bar{\lambda}^k + \mu(\bar{w}^{k+1} - \nabla \phi^{k+1}).$$
 - 2.5. ϕ^{k+1} needs to be processed by thresholding:

$$\phi^{k+1} = \begin{cases} 1 & \phi^{k+1} > \alpha \\ 0 & \text{otherwise} \end{cases}$$
3. The overall loop will be terminated if the stopping criterions (described in Termination Criteria) are satisfied.

For Subproblem 1 the unknown intensities u_1 and u_2 can be obtained by the conditions:

$$u_1^{k+1} = \frac{\int_{\Omega} f \phi^k dx}{\int_{\Omega} \phi^k dx} \text{ and } u_2^{k+1} = \frac{\int_{\Omega} f(1-\phi^k) dx}{\int_{\Omega} (1-\phi^k) dx}.$$

For Subproblem 2 dealing with the iterations of φ , we can have:

$$\begin{aligned} \tilde{\phi}^{k+1} &= \Re \left(IFFT \left(\frac{FFT(g)}{-2\mu(\cos Z_i + \cos Z_j - 2)} \right) \right), \\ \phi^{k+1} &= \prod_{\phi}(\tilde{\phi}^{k+1}) = \max(\min(\tilde{\phi}^{k+1}, 1), 0) \end{aligned} \quad (3.49)$$

where $g = -\mu(\partial_x^- w_1^k + \partial_y^- w_2^k) - (\partial_x^- \lambda_1^k + \partial_y^- \lambda_2^k) - r(u_1^{k+1}, u_2^{k+1})$, and the projection $\prod_{\phi}(\cdot)$ is a simple truncation of $\tilde{\phi}^{k+1}$ to the interval $[0, 1]$ according to [20, 162].

For Subproblem 3 related to calculation of the auxiliary variable \bar{w} , the soft thresholding formulation is used to obtain the result: $\bar{w}^{k+1} = \max(|\nabla \phi^{k+1} - \frac{\bar{\lambda}^k}{\mu}| - \frac{\gamma \varphi'(|\bar{w}|)}{\mu}, 0) \frac{\nabla \phi^{k+1} - \frac{\bar{\lambda}^k}{\mu}}{|\nabla \phi^{k+1} - \frac{\bar{\lambda}^k}{\mu}|}$.

II. NVPM for the Nonlinear Diffusion Based Image Segmentation

As noted earlier, we can reduce the order of energy functional (3.47) and use an alternating optimization method to solve the corresponding subproblems. It has been proved that an alternating optimization method can improve calculation efficiency and reduce the complexity. In this section, NVPM is proposed based on such an idea. It can be optimized using Nesterovs optimal first-order methods [63] for accelerating the iterations. First, we

introduce the computational procedure of NVPM. From the Euler-Lagrange equation of the functional (3.47):

$$r(u_1, u_2) - \gamma \nabla \cdot \left(\varphi'(|\nabla \phi|) \frac{\nabla \phi}{|\nabla \phi|} \right) = 0, \quad (3.50)$$

It is very difficult to calculate the normal vector $\frac{\nabla \phi}{|\nabla \phi|}$, and the high-order term $\nabla \cdot \left(\varphi'(|\nabla \phi|) \frac{\nabla \phi}{|\nabla \phi|} \right)$ is believed to result in slow computation. Here we introduce an auxiliary vector $\vec{p} = [p_1, p_2]^r$ such that $\vec{p} \approx \frac{\nabla \phi}{|\nabla \phi|}$ according to the concept of replacing high-order terms with low-order terms to reduce the order of the variables. Thus the solving process can be simplified and then alternating optimization method is selected to calculate variables as mentioned before. There are three separate subproblems for u, ϕ and \vec{p} . The detailed implementation of NVPM is shown as follows.

NVPM for the nonlinear diffusion based image segmentation

1. Initialization: $u^0 = 0, \phi^0 \in \{0, 1\}, \vec{p}^0 = 0$.
 2. For $k = 0, 1, \dots$ solving the following problems alternatively:
 - 2.1. Subproblem 1 about u^{k+1} :

$$u^{k+1} = \operatorname{argmin}\{\varepsilon_1(u) = E(u, \phi^k)\}.$$
 - 2.2. Subproblem 2 about ϕ^{k+1} :

$$\phi^{k+1} = \operatorname{argmin}\{\varepsilon_2(\phi) = E(u^{k+1}, \phi)\}_{\vec{p} \approx \frac{\nabla \phi}{|\nabla \phi|}}.$$

$$\phi^{k+1} = \prod_{\phi}(\phi^{k+1}).$$
 - 2.3. Subproblem 3 about \vec{p}^{k+1} :

$$\vec{p}^{k+1} = \Pi_{\vec{p} \approx \frac{\nabla \phi}{|\nabla \phi|}}(\phi^{k+1}).$$
 - 2.4. ϕ^{k+1} needs to be processed by thresholding:

$$\phi^{k+1} = \begin{cases} 1 & \phi^{k+1} > \alpha \\ 0 & \text{otherwise} \end{cases}$$
 3. The overall loop will be terminated if the stopping criterions (Termination Criteria) are satisfied.
-

The normal vector projection $\Pi_{\vec{p} \approx \frac{\nabla \phi}{|\nabla \phi|}}(\cdot)$ is used to constrain the instrumental vector \vec{p} . For $k = 0, 1, \dots$ the minimizers of variables u, ϕ and \vec{p} in subproblems 1-3 can be obtained by minimizing the energy functionals and the projection method:

$$\begin{cases} \varepsilon_1(u) = \int_{\Omega} r(u_1, u_2) \phi^k dx \\ \varepsilon_2(\phi) = \int_{\Omega} r(u_1^{k+1}, u_2^{k+1}) \phi dx + \gamma \int_{\Omega} \varphi(|\nabla \phi|) dx \\ \vec{p}^{k+1} = \Pi_{\vec{p} \approx \frac{\nabla \phi}{|\nabla \phi|}}(\phi^{k+1}) \end{cases} \quad (3.51)$$

In the following part, equations given in (3.51) will be solved, respectively.

Minimization of $\varepsilon_1(u)$ in (3.51) u_1 and u_2 can be obtained as $u_1^{k+1} = \frac{\int_{\Omega} f\phi^k dx}{\int_{\Omega} \phi^k dx}$ and $u_2^{k+1} = \frac{\int_{\Omega} f(1-\phi^k) dx}{\int_{\Omega} (1-\phi^k) dx}$.

Minimization of $\varepsilon_2(\varphi)$ in (3.51) The variables u^{k+1} and \vec{p}^k are fixed to calculate ϕ . By introducing the instrumental vector \vec{p} , the Euler-Lagrange equation (3.50) can be simplified as follows:

$$r(u_1^{k+1}, u_2^{k+1}) - \gamma \nabla \cdot (\varphi'(|\nabla \phi|) \vec{p}^k) = 0. \quad (3.52)$$

If this Euler-Lagrange equation is used to obtain ϕ^{k+1} directly, the problem of convergence difficulty or un-convergence will come into the energy curve. Thus, the gradient descent method is added in order to achieve the absolute convergence as:

$$\begin{cases} \frac{\partial \phi}{\partial t} = \gamma \nabla \cdot (\varphi'(|\nabla \phi|) \vec{p}^k) - r(u_1^{k+1}, u_2^{k+1}) & \text{in } \Omega \\ \frac{\partial \phi}{\partial \vec{n}} = 0 & \text{on } \partial \Omega. \end{cases} \quad (3.53)$$

where the second formula of (3.53) is the boundary condition. After obtaining the value of ϕ^{k+1} , the same projection method is used (as in (3.49)) to make sure $\phi^{k+1} \in [0, 1]$.

Projection on \vec{p} in (3.51) It is easy to calculate \vec{p}^{k+1} according to (3.51) under the condition of obtaining u^{k+1} and ϕ^{k+1} . Here the norm vector projection can be directly used as:

$$\vec{p} = \frac{\nabla \phi}{|\nabla \phi|}. \quad (3.54)$$

The numerical problem in above term is mainly caused by its singularity and nondifferentiability. As described in [180], this term is usually approximated as $\vec{p} = \frac{\nabla \phi}{|\nabla \phi|} = \frac{\nabla \phi}{\sqrt{|\nabla \phi|^2 + \varepsilon}}$ (ε is a small positive constant to avoid zero division). Thus the range of \vec{p}^{k+1} can be obtained by the nature of the approximated norm vector as follows:

$$\vec{p}^{k+1} = \begin{cases} 0 & |\nabla \phi^{k+1}| = 0 \\ \frac{\nabla \phi^{k+1}}{|\nabla \phi^{k+1}|} = \frac{\nabla \phi}{\sqrt{|\nabla \phi|^2 + \varepsilon}} & 0 < |\nabla \phi^{k+1}| \leq 1 \\ \frac{\nabla \phi^{k+1}}{|\nabla \phi^{k+1}|} = \frac{\nabla \phi}{\sqrt{|\nabla \phi|^2}} & |\nabla \phi^{k+1}| > 1. \end{cases} \quad (3.55)$$

Using the constraint condition $|\vec{p}^{k+1}| \leq 1$ can summarize above nature, the following method

can guarantee this constraint condition.

$$\vec{p}^{k+1} = \frac{\nabla\phi^{k+1}}{\max(|\nabla\phi^{k+1}|, 1)}. \quad (3.56)$$

which is a simple re-projection method, which seems to perform better. At the end of the implementation, the binarization of ϕ^{k+1} is required as well.

III. Developed version of NVPM: NVPM*

In this part, we will discuss how to optimize the normal vector projection method. From NVPM, it can be seen that the key factor influencing the rate of convergence is the use of gradient descent method. As described in [63], the solution process of Subproblem 2 in NVPM can be accelerated through the over-relaxation iterative method. After obtaining the piecewise constant parameters u_1^{k+1} and u_2^{k+1} , the over-relaxation parameter α and intermediate variable $\tilde{\phi}$ are introduced. ϕ should be calculated by the intermediate variable as:

$$\phi^{k+1} = \tilde{\phi}^k + t(\gamma\nabla \cdot (\varphi'(|\nabla\tilde{\phi}^k|)\vec{p}^k) - r(u_1^{k+1}, u_2^{k+1})). \quad (3.57)$$

Then the over-relaxation parameter α is updated (set $\alpha^0 = 1$):

$$\alpha^{k+1} = \frac{1 + \sqrt{4(\alpha^k)^2 + 1}}{2}. \quad (3.58)$$

The $(k+1)$ th and k th values of ϕ , α are used to update the intermediate variable $\tilde{\phi}$:

$$\tilde{\phi}^{k+1} = \phi^{k+1} + \frac{\alpha^k - 1}{\alpha^{k+1}}(\phi^{k+1} - \phi^k). \quad (3.59)$$

The rest of the steps will not be repeated because they are similar to (3.54)-(3.56) in NVPM.

Termination Criteria

The iterations need to be terminated when the criterions are satisfied. For details, please refer to the termination criteria section given in Section 3.2.2 or [161].

NVPM has more concise algorithm framework and guarantees fast convergence of the energy functional in theory. Therefore, it can greatly enhance the calculation efficiency for

our segmentation models. It could also be a promising scheme to solve other variational problems.

3.5 Experiments

In this section, our proposed models are applied to some synthetic and real images with noise or damage. Experiments are conducted on the same platform (Matlab7.8) on a PC (Intel (R), CPU: 2.60GHz, RAM: 16GB, cores number: 4, architecture: 64-bit).

3.5.1 MTC- ℓ_1 model for color image processing

The ratio (PSNR) is used for performance evaluation and it is defined as

$$PSNR = 10 \log_{10} \frac{mMN \times MAX^2}{\sum_{i=1}^m \|f_i - u_i\|_{\ell_2}^2}. \quad (3.60)$$

where m stands for the layers of the color image. M and N are the height and width, respectively, of the original image. MAX is 255. \vec{u} is the restored image, and \vec{f} is the original image. $\|\cdot\|_{\ell_2}$ denotes the ℓ_2 norm on image domain.

Experiments for image de-noising

A synthetic image of size 256×256 , the popular Lena image of size 200×200 , and a real image of a building of size 237×237 , are used as the test images for denoising. In these experiments, the MTV- ℓ_1 model is used for comparison in order to show that the proposed model can simultaneously improve the staircase effect while keep corners. First, we set $\vec{u}^0 = \vec{f}$, $\vec{q}^0 = \vec{v}^0 = \vec{0}$, $\vec{p}^0 = \vec{m}^0 = \vec{n}^0 = \vec{0}$. and all the Lagrange multipliers are initially set to be 0 for all the numerical experiments. The same initiations of variables in each experiment are used in order to have a relatively neutral criterion. In this experiment, the parameters for our proposed model are given as:

Fig. 3-1(a3): $\lambda = 10$, $\mu_1 = 3 \cdot 10^{-3}$, $\mu_2 = 0.3$, $\mu_3 = 2 \cdot 10^{-3}$, $\mu_4 = 0.2$, $\mu_5 = 0.01$;
 Fig. 3-1(b3): $\lambda = 1$, $\mu_1 = 0.3$, $\mu_2 = 0.02$, $\mu_3 = 2 \cdot 10^{-3}$, $\mu_4 = 2$, $\mu_5 = 0.05$;
 Fig. 3-1(c3): $\lambda = 1.6$, $\mu_1 = 8 \cdot 10^{-3}$, $\mu_2 = 0.04$, $\mu_3 = 0.02$, $\mu_4 = 0.01$, $\mu_5 = 0.01$;

In Fig. 3-1, results of the MTV- ℓ_1 model and the proposed model are first presented, respectively. Fig. 3-1(a1), (b1) and (c1) show the noisy images. The Pepper and salt noise

with the standard deviation 0.1 is used in (a1) and (b1), and the Gaussian noise with the standard deviation 0.01 is used in (c1). The restored images by using the MTV- ℓ_1 model are shown in Fig. 3-1(a2), (b2) and (c2). Fig. 3-1(a3), (b3) and (c3) show the results of our proposed model. The residual images (Fig. 3-1(a4), (a5), (b7) and (b8)) and the plots of horizontal intersections (Fig. 3-1(a6), (a7), (b9) and (b10)) are also given. To see the detailed differences, a small block is cropped from the test images as shown in Fig. 3-1(b4)(b6) and (c4)(c6). From Fig. 3-1(a2) and the residual image (a4), it can be observed that the corners of the star and triangle are smeared using the MTV- ℓ_1 model, but they are preserved well using the proposed model as shown in Fig. 3-1(a3). This feature can be seen clearly in Fig. 3-1(a5) and (a7). Moreover, the color of the objects in Fig. 3-1(a3) and the darkness in Fig. 3-1(c3) look more similar to that in the original noisy images. It demonstrates that our proposed model can keep the contrast invariant, i.e., the loss of image contrast does not exist. The horizontal intersection obtained by randomly choosing three slices with the same position in the noisy, original and result images is another perspective to check the de-noising results. Compared with Fig. 3-1(a6), (a7) shows that the de-noising intersection curve (red) in MTC- ℓ_1 almost overlaps with the original intersection curve (blue), which validates that our proposed model performs better than MTV- ℓ_1 . Fig. 3-1(b8) and (b10) shows that our model can restore more details for more complex images. Comparisons of small regions shown in Fig. 3-1 demonstrate that our model is capable of improving the staircase effect quite well. The parameters used for MTV- ℓ_1 model are chosen as $\lambda = 9$ for Fig. 3-1(a2), $\lambda = 7$ for (b2) and $\lambda = 3$ for (c2), respectively.

From left to right, we can illustrate the relative residuals (3.32), the relative errors of Lagrange multipliers (3.33), the relative error of u^k and the energy curve along the outer layer repeat k (3.34) in Fig. 3-2. These graphs are obtained from Fig. 3-1(a3) and (b3), respectively. Other graphs from Fig. 3-1(c3) have similar profiles. In this section, we show the original data and provide the results for 100 iterations. From these plots, it can be observed that the proposed algorithm converges well before 100 iterations. They also give important information about how to choose penalty parameters μ_i ($i = 1, 2, \dots, 5$). In order to guarantee the convergence as well as the speed of convergence, the constraint errors R_i^k should converge to zero with nearly the same speed. If R_i^k goes to zero faster than others, then μ_i should be decreased. R_i^k will converge to zero with the same speed as the iteration proceeds and the energy will decrease to a steady constant value when μ_i are

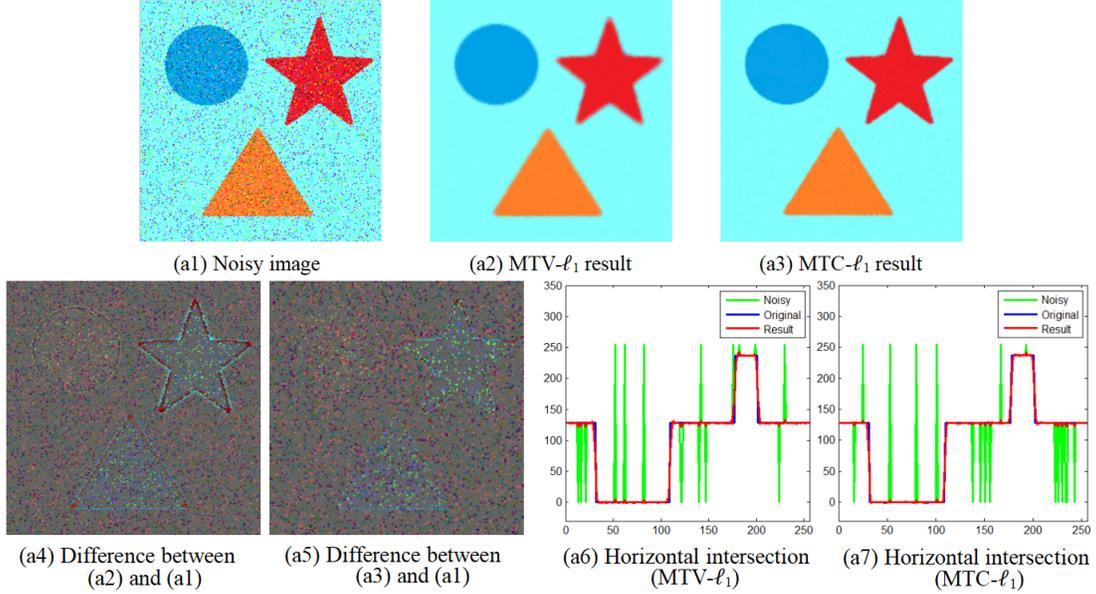


Figure 3-1: The denoising effects of $MTV-\ell_1$ model and our proposed $MTC-\ell_1$ model.

chosen properly.

In regard the efficiency of the proposed algorithm, the energy curves using GDM and ADMM for our proposed model are depicted in Fig. 3-3. As in the above case, Fig. 3-3(f) and (g) are obtained from Fig. 3-1(a3) and (b3), respectively. It is easy to see that the ADMM has much faster convergence rate and higher efficiency in comparison with the GDM.

Table 3.1 shows the comparisons of PSNR values and computation time using different models. The number of iterations is the number of total outer iterations in Fig. 3-1. The computational time is measured in seconds. It can be seen that the proposed model can achieve much better PSNR values with similar time costs.

Table 3.1: Comparisons of PSNR, iterations and computational time.

Models	PSNR (de-noising)	Iterations	Time (s)
Fig. 3-1(a2): $MTV-L1$	27.067	56	6.573
Fig. 3-1(a3): $MTC-L1$	28.660	55	6.688
Fig. 3-1(b2): $MTV-L1$	24.508	51	4.654
Fig. 3-1(b3): $MTC-L1$	28.136	48	5.191
Fig. 3-1(c2): $MTV-L1$	19.845	20	3.154
Fig. 3-1(c3): $MTC-L1$	24.231	22	3.644

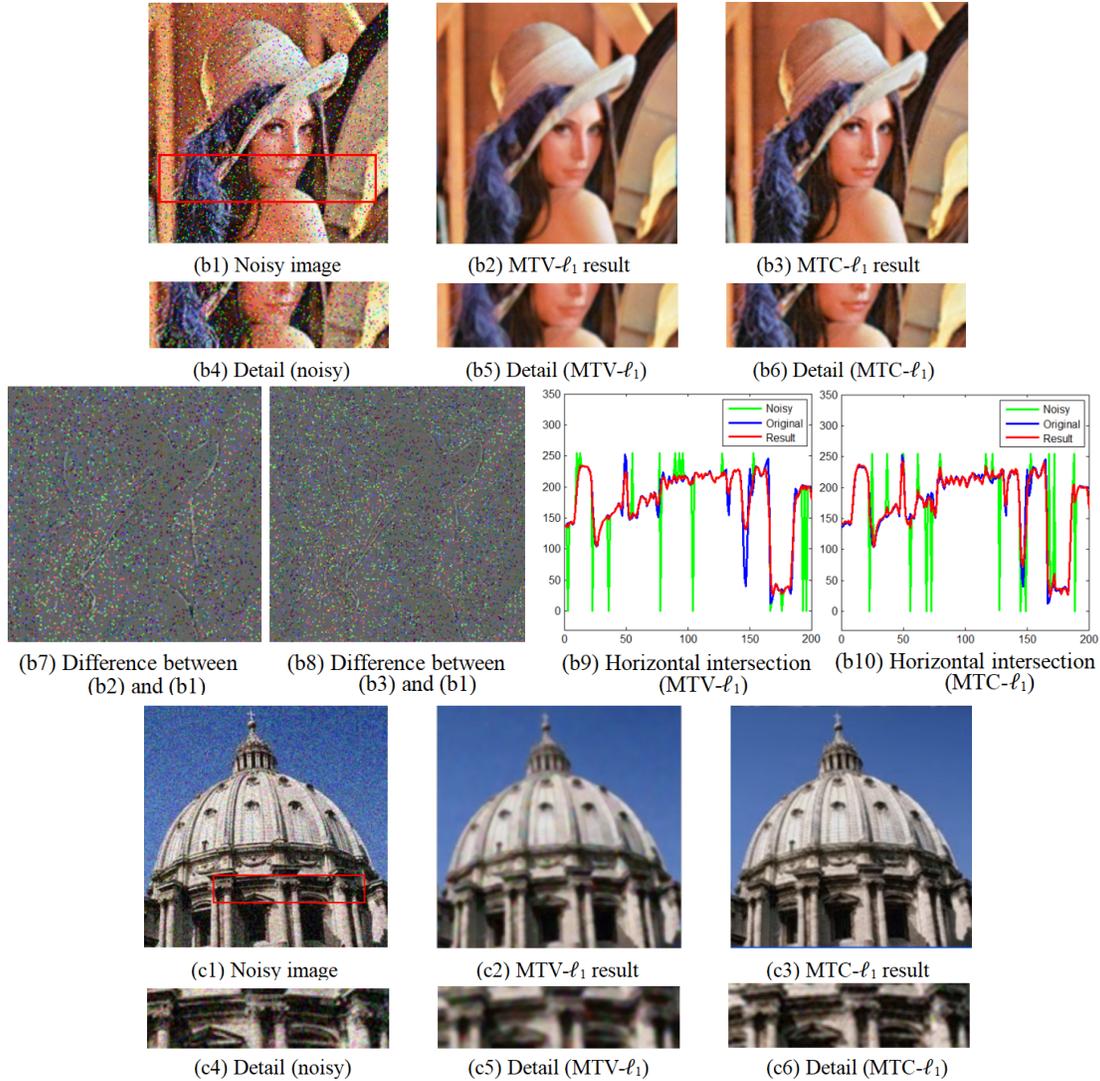


Figure 3-1 continued.

Experiments for image inpainting

A synthetic image of size 256×256 , a parrot image of size 310×310 and a flower image of size 309×255 are used as the test images for inpainting. The red regions in the first, second and third rows are the inpainting domains. In these experiments, visual comparisons can be made with the inpainted images subsequently. The damage types are marked by arbitrary color paintings to illustrate that our proposed model can inpaint effectively and the micro structures in feather and wing of the parrot show the benefit of our proposed model in edge

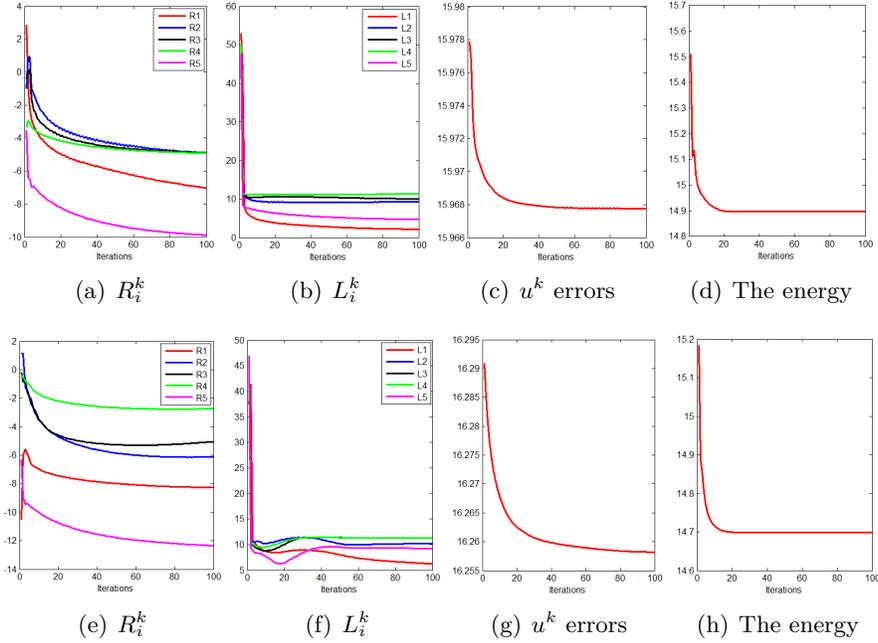


Figure 3-2: The plots of parametric errors and the energy curve. (a-d) are obtained by our MTC- ℓ_1 model from Fig. 3-1(a3). (e-h) are obtained by our MTC- ℓ_1 model from Fig. 3-1(b3).

preserving. The model to solve the image inpainting problem is stated as

$$E(\vec{u}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |\kappa_i|^2} dx + \sum_{i=1}^m \int_{\Omega} \lambda_D |u_i - f_i| dx, \quad (3.61)$$

where $\lambda_D = \begin{cases} 0 & x \in D \\ \lambda & x \in \Omega/D \end{cases}$ is the mask function to represent the broken region. The algorithm for this inpainting model is the same as the one mentioned in Sections 3.4.1. Notice that \vec{v}^{k+1} should be obtained through the following formula in this case,

$$v_i^{k+1} = \max \left(\left| u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5} \right| - \frac{\lambda_D}{\mu_5}, 0 \right) \frac{u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5}}{\left| u_i^{k+1} - f_i - \frac{\lambda_{i5}^k}{\mu_5} \right|}. \quad (3.62)$$

The parameters used to obtain Fig. 3-4(a3) are shown as follows:

-
- Fig. 3-4(a2): $\lambda = 3$, $\mu_1 = 4 \cdot 10^{-3}$, $\mu_2 = 3 \cdot 10^{-3}$, $\mu_3 = 0.02$, $\mu_4 = 6 \cdot 10^{-3}$, $\mu_5 = 7 \cdot 10^{-3}$;
 Fig. 3-4(b2): $\lambda = 2$, $\mu_1 = 2 \cdot 10^{-3}$, $\mu_2 = 4 \cdot 10^{-3}$, $\mu_3 = 2 \cdot 10^{-3}$, $\mu_4 = 0.02$, $\mu_5 = 0.02$;
 Fig. 3-4(c2): $\lambda = 2$, $\mu_1 = 2 \cdot 10^{-3}$, $\mu_2 = 0.02$, $\mu_3 = 0.03$, $\mu_4 = 0.02$, $\mu_5 = 0.03$;
-

In Fig. 3-4, some results of the MTV- ℓ_1 model and our proposed MTC- ℓ_1 model are shown, respectively. Fig. 3-4(a1), (b1) and (c1) shows the damaged images. The restored

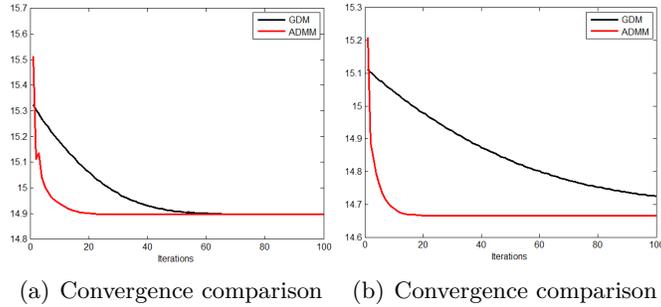


Figure 3-3: Comparisons of energy curves. (a): energy curves of using GDM and ADMM for example shown in Fig. 3-1(a3); (b): energy curves of using the GDM and ADMM from Fig. 3-1(b3).

images by using the MTV- ℓ_1 are shown in Fig. 3-4(a2), (b2) and (c2). Fig. 3-4(a3), (b3) and (c3) shows the results of our proposed model. The differences can be better observed by cropping a small block from test images as shown in Fig. 3-4(b4)(b6). It can be seen that our proposed model performs particularly well in inpainting and the results are visually much better. The parameters used for MTV- ℓ_1 model are chosen as $\lambda = 9$ for Fig. 3-4(a2), $\lambda = 7$ for (b2) and $\lambda = 3$ for (c2), respectively.

Fig. 3-5 shows the relative residuals (3.32), the relative errors of Lagrange multipliers (3.33), the relative error of u^k and the energy curve along the outer repeat k (3.34). Only the first 100 iterations are shown, and the plots in this section are shown in log-scale. The convergence rates are the same as the original data. Fig. 3-4(a3) provides the information for the graphs.

The energy curves by using the GDM and ADMM for our proposed model are given in Fig. 6. It validates that the alternating direction method of multipliers has a significant advantage in computational efficiency.

In Table 3.2, comparisons of PSNRs and computation time using different models are shown. From the information in this table, it is obvious that our proposed model can achieve much better performance in image inpainting.

MTC- ℓ_2 model compared with our proposed model

Experiments in terms of the scale spaces generated by MTC- ℓ_2 model and our MTC- ℓ_1 model on a real image are given in Fig. 3-7. The energy functional of MTC- ℓ_2 is shown as

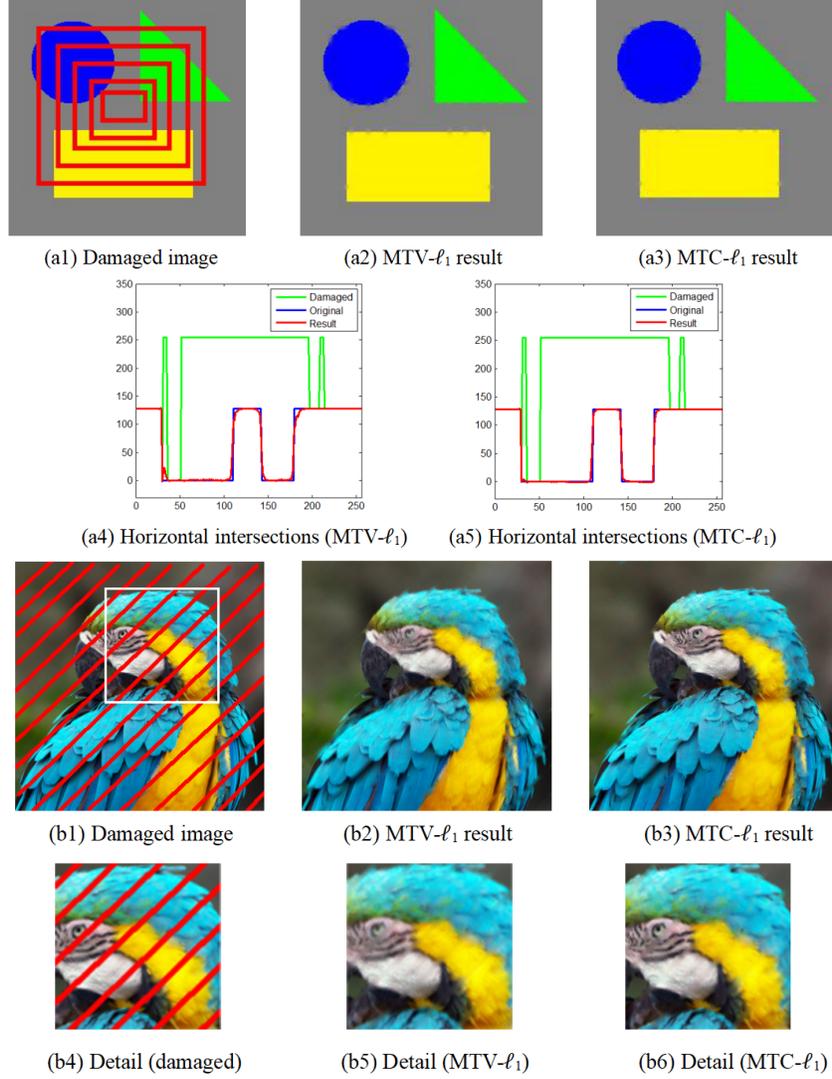


Figure 3-4: The inpainting effects of MTV- ℓ_1 model and our proposed MTC- ℓ_1 model.

follows:

$$E(\vec{u}) = \int_{\Omega} \sqrt{\sum_{i=1}^m |\kappa_i|^2} dx + \lambda \sum_{i=1}^m \int_{\Omega} (u_i - f_i)^2 dx. \quad (3.63)$$

Fig. 3-7(a1) and (b1) give the original Lena images of size 200×200 . Different values for parameter λ are set for these two models to see which one is better in contrast preserving. (a2)(a6) show results of MTC- ℓ_2 model with scale space from $\lambda = 3$ to $\lambda = 0.05$. (b2)(b6) show results of our proposed MTC- ℓ_1 model with bigger scale space from $\lambda = 10$ to $\lambda = 0.001$. When scale changes, it can be observed obviously that the ℓ_1 fidelity model maintains their contrast much better, especially very small values for λ are used. For example, $\lambda = 0.1$:

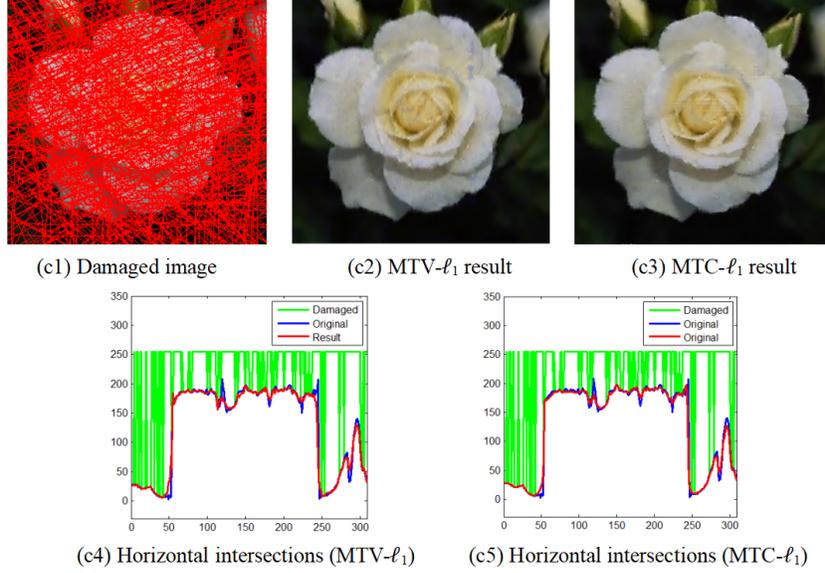


Figure 3-4 continued.

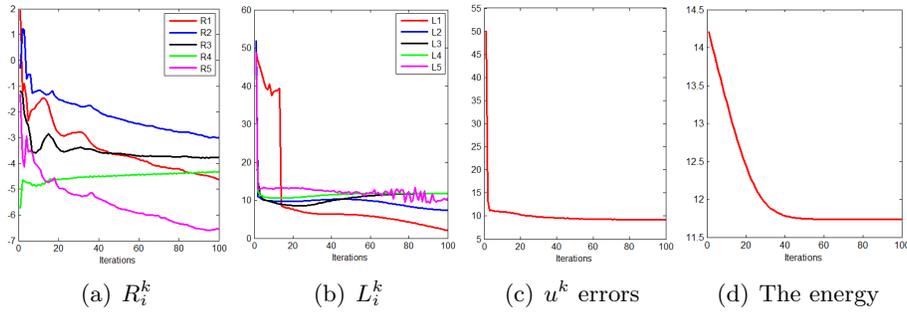
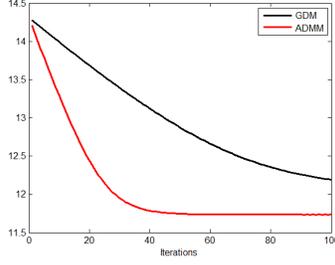


Figure 3-5: The plots of parametric errors and the energy curve. These plots are obtained by our $MTC-\ell_1$ model for image inpainting from Fig. 3-4(a3).

our model can produce very good result; $\lambda = 0.05$: the general outlines can be seen by our model. Even though the parameter λ is gradually decreased to very low values, our model still shows a stronger performance in contrast preserving.

Layered total curvature model based on ℓ_1 norm ($LTC-\ell_1$) compared with our proposed $MTC-\ell_1$ model

In this section, we apply the total curvature to the layered color model and compare it with the proposed model. Visual comparisons with the results are shown in Fig. 3-8 and it can be observed that the results of two methods are very close as evidenced from Table 3.3. The main idea of $LTC-\ell_1$ model is to apply $TC-\ell_1$ model on each channel, which can be



(a) Convergence comparison

Figure 3-6: Comparisons of energy curves. (e): energy curves of using the GDM and ADMM for example shown in Fig. 3-4(a3).

Table 3.2: Comparisons of PSNR, iterations and computational time.

Models	PSNR (inpainting)	Iterations	Time (s)
Fig. 3-4(a2): MTV-L1	32.923	60	3.159
Fig. 3-4(a3): MTC-L1	36.219	35	4.853
Fig. 3-4(b2): MTV-L1	26.003	59	4.564
Fig. 3-4(b3): MTC-L1	27.412	42	5.671
Fig. 3-4(c2): MTV-L1	29.631	43	4.119
Fig. 3-4(c3): MTC-L1	31.211	25	5.064

stated as

$$E(\vec{u}) = \sum_{i=1}^m \int_{\Omega} |\kappa_i| dx + \lambda \sum_{i=1}^m \int_{\Omega} |u_i - f_i| dx. \quad (3.64)$$

The parameters used in our model for Fig. 3-8:

$$\text{Fig. 3-8(a3): } \lambda = 15, \mu_1 = 1 \cdot 10^{-4}, \mu_2 = 0.5, \mu_3 = 0.3, \mu_4 = 0.3, \mu_5 = 0.1;$$

In Fig. 3-8, some results of LTC- ℓ_1 model and the proposed MTC- ℓ_1 model are shown, respectively. Fig. 3-8(a1) gives the noisy image. The restored images are shown in Fig. 3-8(a2) and (a3). To see the detailed differences, we crop a small block from the test images as shown in Fig. 3-8(a4)(a6). Magnifications of the small block and histograms show that LTC- ℓ_1 model may cause some blurred edges.

In Table 3.3, the comparisons of PSNR values and computation time by using different models are shown. It can be seen that these two methods are similar in performance and computing time. The only difference is that our model does better in coupling effect with slightly higher PSNR value. The parameters used for LTC- ℓ_1 model are chosen as $\lambda = 11$ for Fig. 3-8(a2).

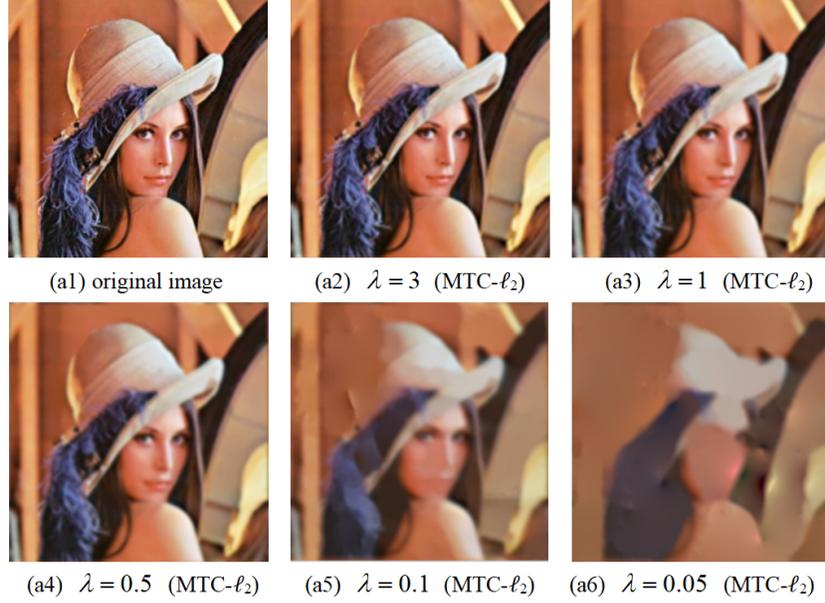


Figure 3-7: Scale spaces generated by $MTC-\ell_2$ model and our $MTC-\ell_1$ model.

Table 3.3: Comparisons of PSNR, iterations and computational time.

Models	PSNR (de-noising)	Iterations	Time (s)
Fig. 3-8(a2): LTC-L1	28.869	52	5.926
Fig. 3-8(a3): MTC-L1	29.449	35	4.688

3.5.2 Illusory Contour Capture

Experiments: Illusory Contour Capture on Synthetic Images

Some synthetic images of size 115×115 pixels containing a triangular and round objects are used as test images in this section. Zhus original model (2.23) is used for comparison in order to show the performance of our proposed model. Experiments validated that our simplified model as shown in Eq. (3.35) can produce similarly good results as Zhus (2.23), while our algorithm shows much better efficiency due to its simpler framework with easy calculations. The time costs of using Zhus model (2.23) via GDM and our model (3.35) via ADMM-P are compared in the experiments. \vec{w} , v and all the Lagrange multipliers are initially set to be 0 for all the numerical experiments. They are initialized the same way to ensure a relatively fair comparison. The detailed implementation framework is shown in Fig. 3-9 and all experiments follow the same procedure.

In Fig. 3-10, the original images are shown in (a1) and (b1), and results obtained by a standard segmentation approach in [40] are presented in (a2) and (b2). (a3) and (b3) give

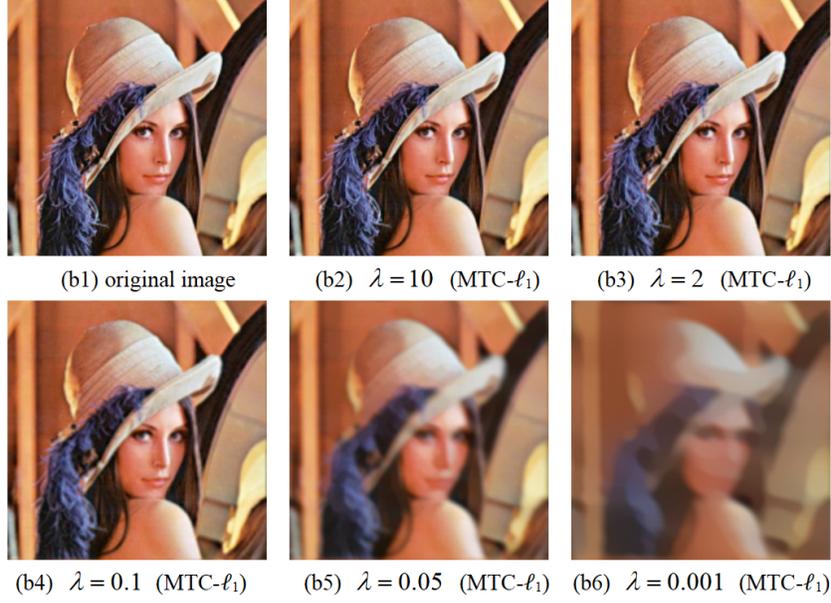


Figure 3-7 continued.

the final results obtained by Zhus model (2.23), results from our model (3.35) are presented in (a4) and (b4). It can be clearly seen that the results obtained by our simplified model are nearly the same as those by Zhus model and the results obtained from the standard segmentation [40] (shown in (a2) and (b2)) are totally different since the illusory contours are not connected. Subsequently (a5) and (b5) show the important term d provided in Eq. (2.25) and the initialization φ is shown in (a6) and (b6) for the two synthetic images. Fig. 3-10(c) and (d) show the evolution process of the two approaches, in which (c) is for Zhus intermediate procedure and (d) is for ours. It is clear that our approach is much faster as it converges in 26 steps while Zhus approach needs to run 96 steps to converge. In this experiment, the parameters for Zhus model with the GDM method and our proposed ADMM-P algorithm are set (dt is the time step in GDM) as follow.

GDM method	
the triangular object:	$\mu = 15, \lambda = 2.5, a = 0.1, b = 0.001, dt = 0.2;$
the round object:	$\mu = 13, \lambda = 2, a = 0.1, b = 0.001, dt = 0.4.$
our proposed ADMM-P algorithm	
the triangular object:	$\mu = 0.005, \lambda = 2.5, a = 0.1, b = 0.001, \mu_1 = 5 \cdot 10^{-5}, \mu_2 = 50;$
the round object:	$\mu = 0.005, \lambda = 1.5, a = 0.1, b = 0.001, \mu_1 = 5 \cdot 10^{-5}, \mu_2 = 50.$

To further demonstrate the efficiency of our proposed algorithm, the iteration numbers and computation time using the two models are shown in Table 3.4. It is easy to see that

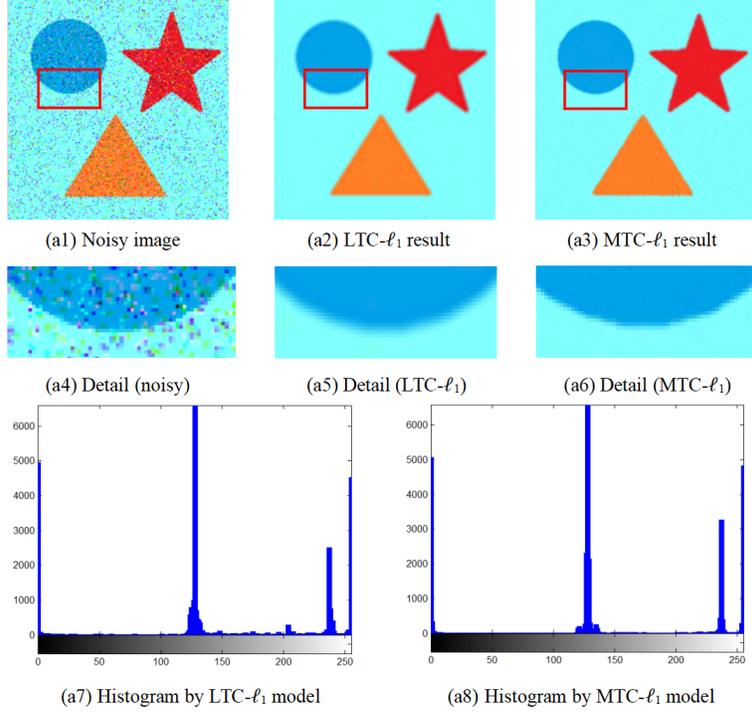


Figure 3-8: The denoising effects of $LTC-\ell_1$ model and our $MTC-\ell_1$ model.

our ADMM-P converges much faster.

Table 3.4: Number of iterations and time costs using GDM and ADMM-P methods.

Images	Approaches	Iterations	Time (s)
Fig. 3-10(a3)	GDM [208]	80	0.35
Fig. 3-10(a4)	ADMM-P	14	0.062
Fig. 3-10(b3)	GDM [208]	96	0.36
Fig. 3-10(b4)	ADMM-P	26	0.095

Experiments: Illusory Contour Capture on Noisy Images

In this section, we apply our model on synthetic noisy images. In Fig. 3-11, (a1) is the clean image (size 100×100), (a2) is the noisy image with the Salt & Pepper noise (standard deviation 0.08), and (a3) shows the standard segmentation result of the noisy image using the method described in [40]. (a4) and (a5) present the results by Zhu's and our model respectively. The foreground of the original image includes two separated parts as shown in (a6) of Fig. 3-11. In Fig. 3-11(a6), the level set of d is shown, followed by the initialization of φ in (a7). It can be seen that the result obtained by our approach is smoother than the one from Zhu's approach. Row (b) and (c) show the different performances between

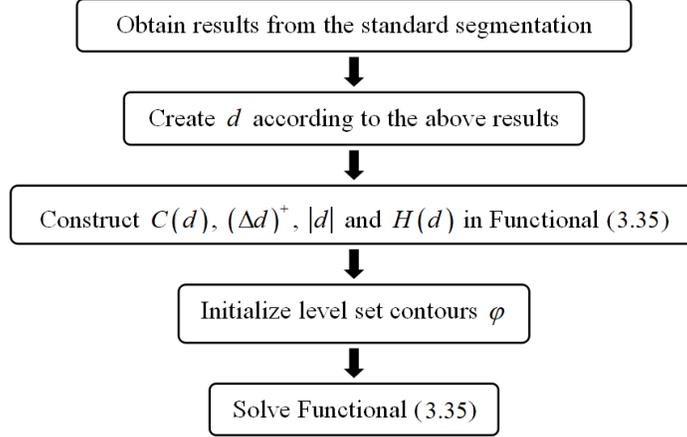


Figure 3-9: The procedure for illusory contour capture.

Zhu’s work and ours with different noise standard deviations (from small to large). It can be observed that our illusory contour capture model is robust to noises and Zhus model cannot achieve ideal results when the image is noisy. Besides, Zhu’s model seems to be more easily affected by the areas with intense noises, while our model is able to obtain smooth reconstruction of the illusory contours on different noise levels. In this experiment, the parameters for the GDM method and our ADMM-P are summarized in the following.

GDM method	
$\mu = 10, \lambda = 2.5, a = 0.1, b = 0.001, dt = 0.2.$	
our proposed ADMM-P algorithm	
$\mu = 0.002, \lambda = 7, a = 0.1, b = 0.001, \mu_1 = 2 \cdot 10^{-3}, \mu_2 = 50.$	

Table 3.5 shows the time costs of Zhus and our models in order to compare their computational efficiency.

Table 3.5: Number of iterations and time costs using GDM and ADMM-P methods.

Images	Approaches	Iterations	Time (s)
Fig. 3-11(a4)	GDM [208]	79	0.35
Fig. 3-11(a5)	ADMM-P	25	0.10

Experiments: Illusory Contour Capture on Real Images (small size)

Next, illusory contour capture on real images containing part of a map (size 266×246), a starfish (size 277×277) and a city square (size 256×256) are tested. Fig. 3-12(a1)-(a4) show the original map image, the result from standard segmentation [40], and the final results

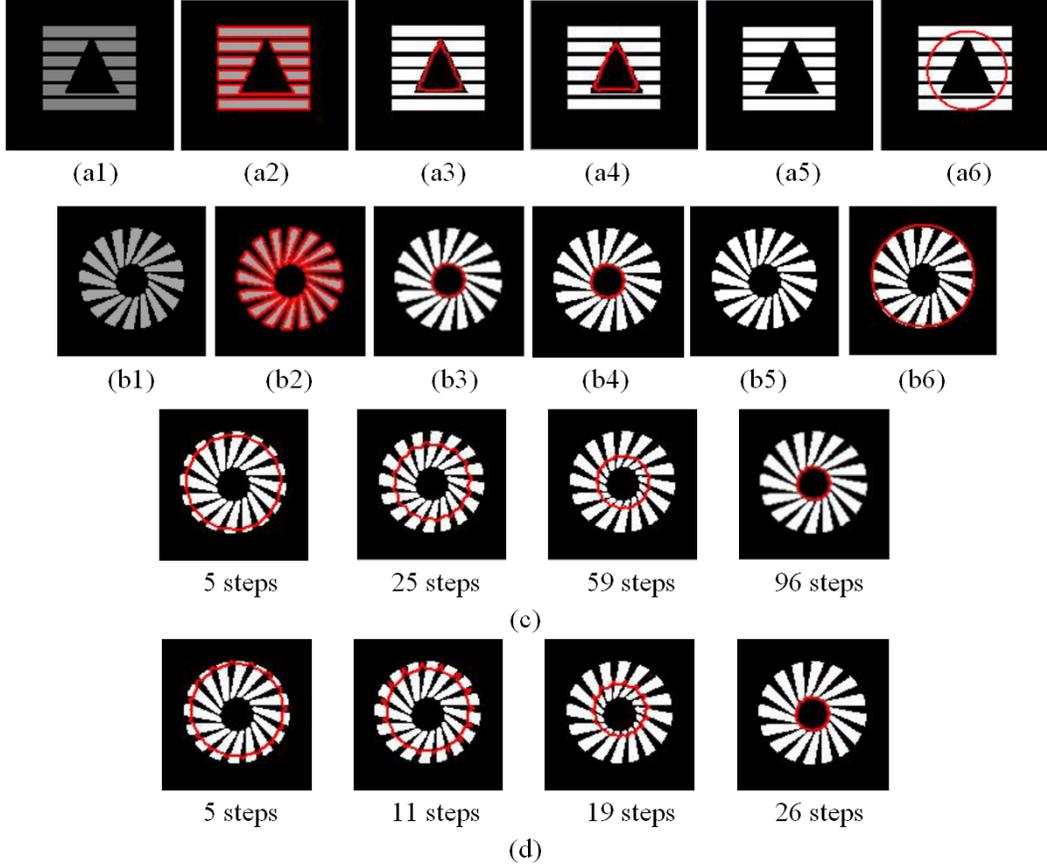


Figure 3-10: Illusory contour capture for synthetic images.

from Zhu's and our models. From left to right, Fig. 3-12(b1)-(b4) and (c1)-(c4) show d of the starfish and square, initialization of φ and the results from Zhu's and our models. Our model can clearly perform well in real images. The parameters for the GDM method and our ADMM-P in this experiment are selected as follows.

GDM method	
map image:	$\mu = 1, \lambda = 4, a = 0.05, b = 0.005, dt = 0.2;$
starfish image:	$\mu = 3, \lambda = 12, a = 0.1, b = 0.003, dt = 0.3;$
square image:	$\mu = 1, \lambda = 8, a = 0.1, b = 0.001, dt = 0.2.$
our proposed ADMM-P algorithm	
map image:	$\mu = 0.005, \lambda = 10, a = 0.1, b = 1 \cdot 10^{-4}, \mu_1 = 8 \cdot 10^{-5}, \mu_2 = 50;$
starfish image:	$\mu = 0.05, \lambda = 10, a = 0.1, b = 0.001, \mu_1 = 5.5 \cdot 10^{-5}, \mu_2 = 50;$
square image:	$\mu = 0.05, \lambda = 10, a = 0.1, b = 0.001, \mu_1 = 6 \cdot 10^{-5}, \mu_2 = 10.$

Table 3.6 shows the time costs using the two methods. It can be seen clearly that similar results have been achieved using the proposed model using much less iterations.

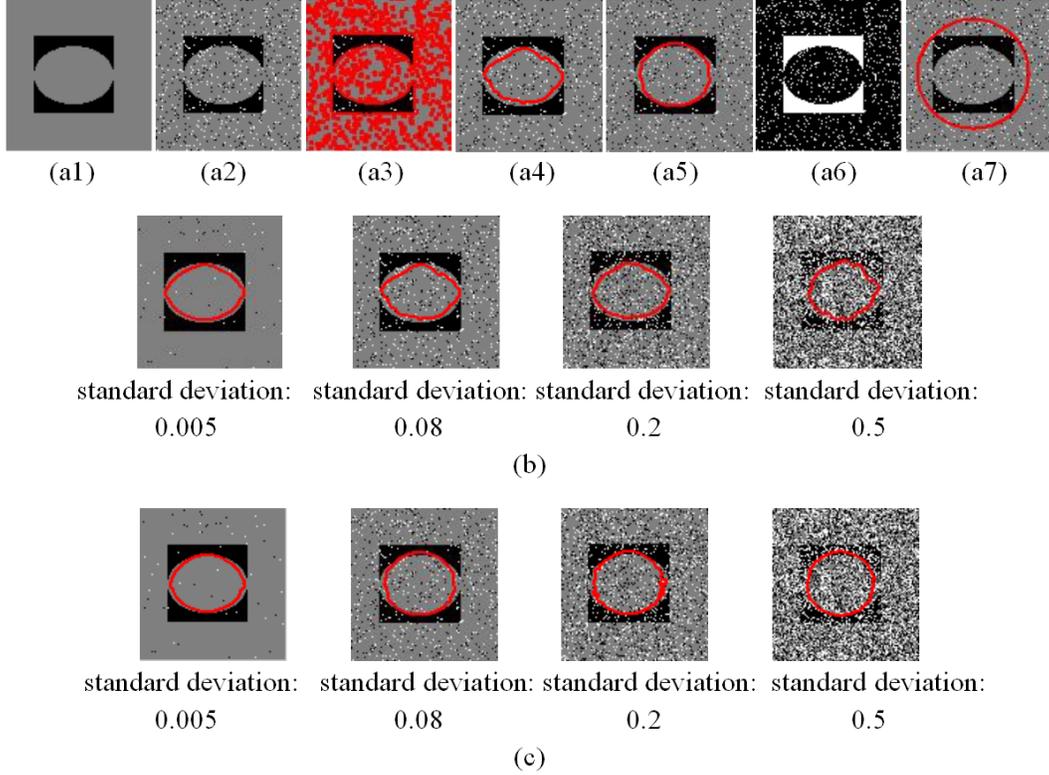


Figure 3-11: Illusory contour capture for noisy images.

Experiments: Illusory Contour Capture on Real Images (large size)

In order to further evaluate the performance of our approach, we apply our model on more complicated images (large, high quality real images) in this section. Visual comparisons of the results are shown in Fig. 3-13. An alpaca image of size 4000×3000 in Fig. 3-13(a1) is used for testing. From our observation, initialization curve roughly enclosing the object can help obtain good results efficiently, which is shown in (a2). The standard segmentation result is shown in (a3). The negative influence of shadows are very clear here. The last two images in (a4) and (a5) show the final segmentation by Zhus and our work respectively.

Table 3.6: Number of iterations and time costs using GDM and ADMM-P methods.

Images	Approaches	Iterations	Time (s)
Fig. 3-12(a3)	GDM [208]	265	1.89
Fig. 3-12(a4)	ADMM-P	56	0.66
Fig. 3-12(b3)	GDM [208]	310	2.03
Fig. 3-12(b4)	ADMM-P	63	0.67
Fig. 3-12(c3)	GDM [208]	178	1.62
Fig. 3-12(c4)	ADMM-P	41	0.59

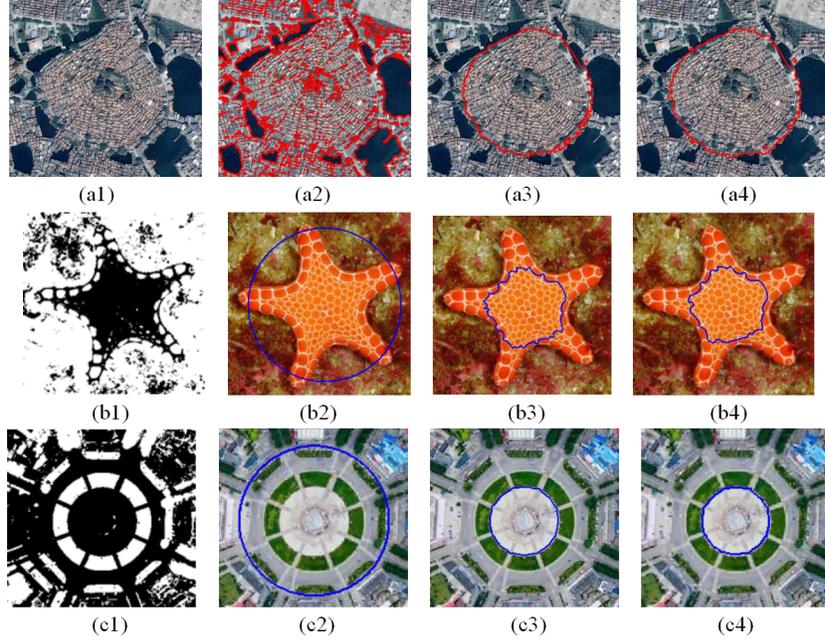


Figure 3-12: Illusory contours capture for three real images.

It can be seen that the performances of two models are similar but as shown in Table 3.7 the advantage of our approach in efficiency is evident. The parameters used for the GDM method and our proposed ADMM-P algorithm are given below.

GDM method	
$\mu = 1, \lambda = 1300, a = 0.05, b = 0.1, dt = 0.2.$	
our proposed ADMM-P algorithm	
$\mu = 0.005, \lambda = 2 \cdot 10^3, a = 0.01, b = 0.1, \mu_1 = 3 \cdot 10^{-4}, \mu_2 = 10.$	

Table 3.7: Number of iterations and time costs using GDM and ADMM-P methods.

Images	Approaches	Iterations	Time (s)
Fig. 3-13(a4)	GDM [208]	52	115.62
Fig. 3-13(a5)	ADMM-P	12	31.04

In general, experiments on synthetic images confirm that our proposed model consumes less time in computation while producing similar or better results. When applying to real and complex images, the time efficiency of our approach becomes more significant. When dealing with noisy images, the benefits of our model become more obvious in terms of accuracy with the increment of noise level. These advantages make our approach more suitable for practical applications.

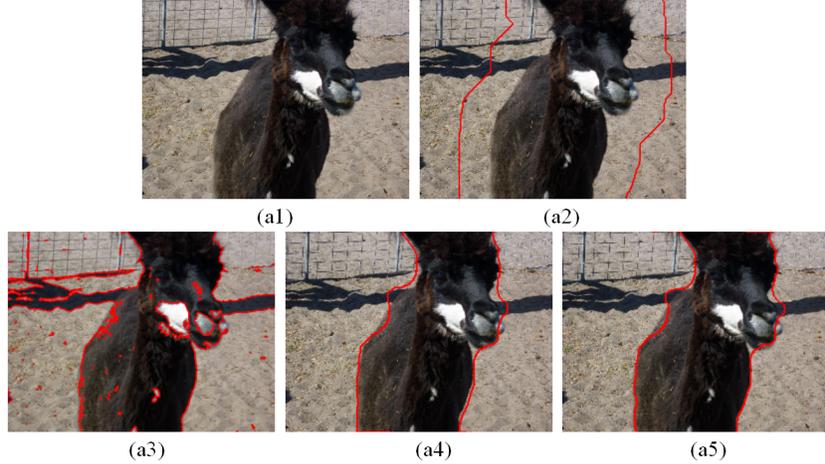


Figure 3-13: Illusory contours capture for large real images.

3.5.3 Nonlinear Diffusion based Image Segmentation

In this section, the numerical results of our proposed model are applied to some grayscale cases (synthetic and real images) and colour cases (real images). We choose the multi-channel Chan-Vese model [37] as the variational framework by means of nonlinear diffusion, and comparisons are done for both edge-based [26] and region-based [171] active contour segmentation models. The same initial contours and initiations of variables for both methods in each experiment are used in order to have a relatively neutral setting for comparison. The initial values of variables are shown as follows:

ADMM: $u^0 = 0, \phi^0 \in \{0, 1\}, \bar{w}^0 = 0, \bar{\lambda}^0 = 0;$
NVPM: $u^0 = 0, \phi^0 \in \{0, 1\}, \bar{p}^0 = 0;$
Optimized NVPM: $t = 0.1, u^0 = 0, \phi^0 \in \{0, 1\}, \tilde{\phi}^0 = \phi^0, \bar{p}^0 = 0.$

Experiments: Segmentation on Grayscale Images

Synthetic images of size 182×178 , 275×275 , 256×256 and 254×254 are used as the test images. In this experiment, our proposed model is applied to images with spatially varying background and textures. Fig. 3-14 shows some results of our model. Geodesic active contour model (GAC model [26]):

$$\frac{\partial \phi}{\partial t} = \mu \nabla \cdot \left(g(x) \frac{\nabla \phi}{|\nabla \phi|} \right) - \beta g(x), \quad (3.65)$$

and, piece-wise smooth, active contour model (PSAC model [171]):

$$\begin{cases} u_1\phi = f\phi + \mu\nabla \cdot (\phi\nabla u_1) \\ u_2(1 - \phi) = f(1 - \phi) + \mu\nabla \cdot ((1 - \phi)\nabla u_2) \\ \frac{\partial\phi}{\partial t} = \gamma\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \alpha_1((u_1 - f)^2 + \mu|\nabla u_1|^2) + \alpha_2((u_2 - f)^2 + \mu|\nabla u_2|^2) \end{cases}, \quad (3.66)$$

are used to compare with our model. The edge function $g(x)$ in (3.65) is defined as $g(x) = \frac{1}{1+|\nabla(G_\sigma * f(x))|^2}$. The equations for obtaining the parameters value u_1 and u_2 of the PSAC model were given in (3.66). According to definition, u_1 and u_2 in (3.66) are in the form of functions while the ones in (2.26) are piece-wise constants. The PSAC model transforms the piece-wise constant parameters u_1 and u_2 into variables that need to be calculated. Fig. 3-14 (a1)-(d1) show the initial curves. The segmentation results of our model are shown in Fig. 3-14 (a2)-(d2). In practice, the results of our model using all the potential functions shown in Table 2.1 are very close. Thus we do not discuss it too much in the following experiments. The initial curves and results of GAC model and PSAC model are shown in Fig. 3-15. The parameters used for each model are shown under each image. From Fig. 3-15 (e2) and (f2), it can be observed that the GAC model cannot detect interior contours of the ring and the box, while the PSAC model cannot detect the objects with varying features as shown in Fig. 3-15 (e3) and (f3). For regions with textures, both GAC and PSAC fail in segmenting the complete regions (Fig. 3-15(g2)-(g3), (h2)-(h3)). In contrast, our model can also successfully get the same results. The initial curves given in Fig. 3-15(e1) and (g1). It does not depend so much on the position of the initial curve. Thus our model can be considered to have better effects in detecting regions with smoothly spatially varying features and textures.

An airplane image of size 414×243 and a squirrel image of size 212×153 are used as the test images in this experiment. Visual comparisons can be made on the segmentation results obtained by our model, GAC model and PSAC model. The same initial contours are used in Fig. 3-16 (a1) and (b1). The results of our model, the GAC model and PSAC model are given from left to right. It can be seen clearly that our proposed model has better segmentation ability and precision on real images.

When dealing with the convex optimization problem, a threshold method has to be used for both ADMM and NVPM to realize the binarization of ϕ^{k+1} . It is an important way to achieve accurate results. Non-threshold solutions of the proposed methods are shown as

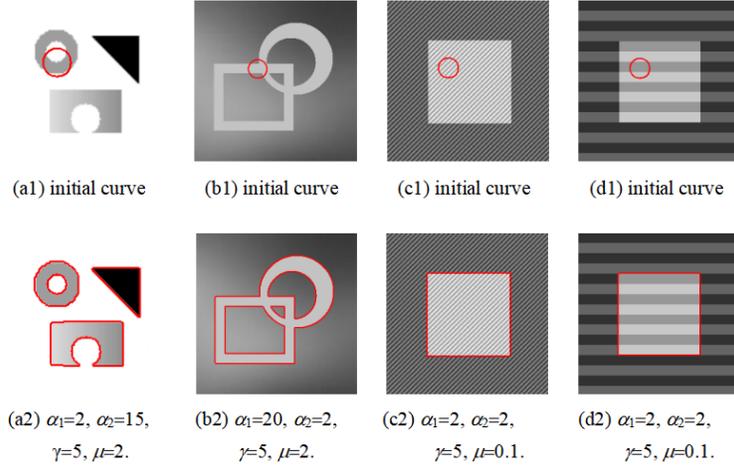


Figure 3-14: Effects of our model.

follows. It can be observed that non-threshold often results in fuzzy regions (red circles in Fig. 3-17 (d2)). In practice, we find the threshold of ϕ^{k+1} could be set at an applicable value of 0.5.

In Table 3.8, comparisons of iterations and computation time using different methods are shown. The number of iterations is the number of total outer iterations. The computational time is measured in seconds. Here PF (ii) and (iv) from Table 2.1 are chosen as examples. It can be seen that the convergence ability of NVPM is nothing less than that of ADMM.

Table 3.8: Comparisons of iterations and time using different methods.

Image	Methods	Iterations	Time (sec)
Fig. 3-16(a2) PF (ii)	ADMM	6	0.184
	NVPM	6	0.178
	NVPM*	6	0.175
Fig. 3-16(b2) PF (iv)	ADMM	16	0.215
	NVPM	9	0.118
	NVPM*	8	0.109

Experiments: Segmentation on Color Image

In this experiment, a bird image of size 300×380 , a leaf image of size 210×280 and an insect image of size 962×642 are used as the test images. Our nonlinear diffusion based

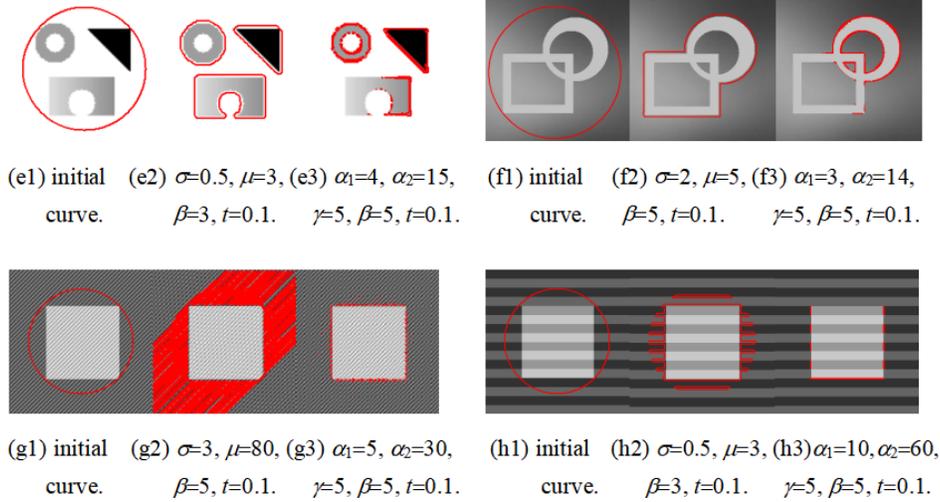


Figure 3-15: Effects of GAC and PSAC model.

multi-channel model is defined as follows:

$$\min_{u, \phi \in [0,1]} \left\{ E(u, \phi) = \alpha_1 \int_{\Omega} \sum_{i=1}^n (u_{1i} - f_i)^2 \phi dx + \alpha_2 \int_{\Omega} \sum_{i=1}^n (u_{2i} - f_i)^2 (1 - \phi) dx + \gamma \int_{\Omega} \varphi(|\nabla \phi|) dx \right\} \quad (3.67)$$

which uses multi-channel Chan-Vese model [37] as the variational framework. The same initial contours for each image using different methods are shown in Fig. 3-18(a1), (b1) and (c1). The segmentation results of our model, the GAC model and the PSAC model are given subsequently. The result obtained by our model using ADMM is presented in Fig. 3-18(a2), (b2) is using the NVPM algorithm framework and (c2) gives the NVPM* result. All the potential functions presented in Table 2.1 can get very similar results. By comparing the results of these models, it is clear that the GAC model can only detect the major objects. It cannot capture detailed object structures (e.g., the birds plumage, the tips of the leaf and the insects legs) and often results in fuzzy edges. The calculation for obtaining the parameters value u_1 and u_2 of the PSAC model were given in (3.66). Parameter values u_1 and u_2 of our multi-channel model (3.67) can be obtained through a similar computation in $\varepsilon_1(u)$ of ADMM and NVPM. It can be observed that the values from (3.66) are in the form of function while the ones from $\varepsilon_1(u)$ of ADMM and NVPM are piece-wise constants. PSAC model transforms the piece-wise constant parameters u_1 and u_2 into variables that

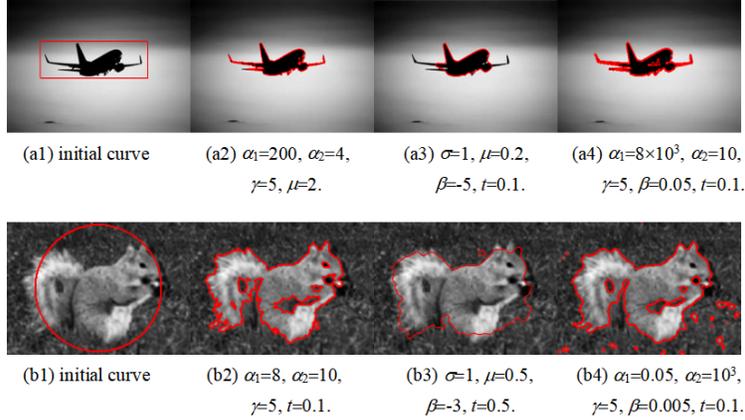


Figure 3-16: Effects of our model, GAC model and PSAC model.

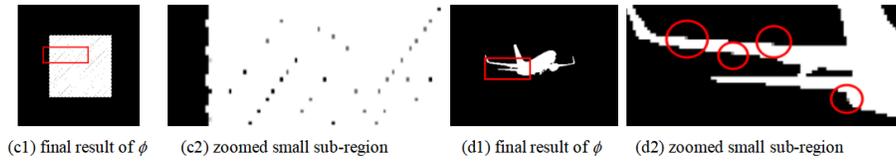


Figure 3-17: Non-threshold solutions of our methods. The first column: results of ϕ . The second column: zoomed small sub-regions (red rectangles in (c1) and (d1)) for comparison.

need to be calculated. Through experiments the performance of PSAC can surpass GAC in general, while inconstant values of u_1 and u_2 included in PSAC may cause the detected objects with excessive details and some tiny structures such as the leaf stem and the insects legs will also be smoothed by the regularizers $|\nabla u_1|^2$ and $|\nabla u_2|^2$.

In Fig. 3-19, plots of relative errors and energy versus iteration numbers of ADMM and NVPM* are shown, for the experiment shown in Fig. 3-18(a2). As mentioned above, all graphs related to our algorithms are drawn up to 50 iterations. By doing so, we can easily observe the convergence of the algorithms. From this test, we can see that NVPM takes fewer iterations to converge compared with ADMM. The two algorithms produce similar results.

At last, the iterations and computational time of these algorithms are compared. Here PF (vi), (viii), (ix) and (x) from Table 2.1 are chosen as examples. For colour image segmentation, the performance achieved is similar to the gray image segmentation. We put the results of this subsection in the summary table in [161].

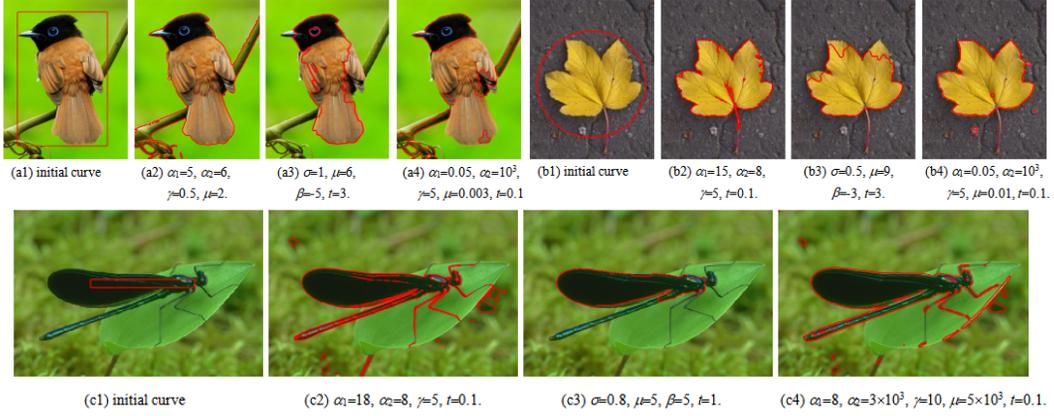


Figure 3-18: Effects of our model, GAC model and PSAC model on colour images.

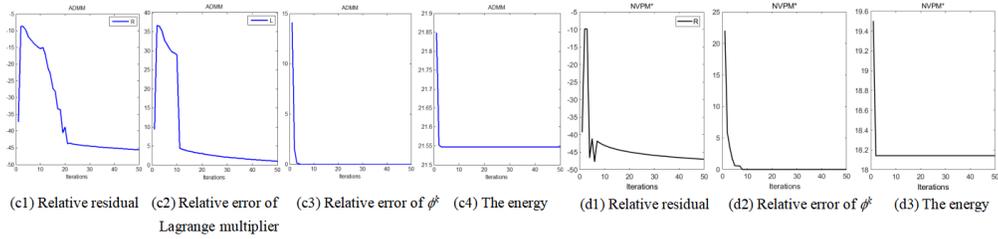


Figure 3-19: Plots of parametric errors and energy curves. The first row is obtained by our model using ADMM. The second row is obtained by our model using NVPM*.

3.6 Summary

The multi-channel total variation (MTV) based on ℓ_2 norm is capable of preserving object edges and smoothing flat regions in color images. However, it will lead to loss of image contrast, smear object corners, and staircase artifacts in the restored images. In order to remedy these side effects, we propose a new multi-channel total curvature model based on ℓ_1 norm (MTC- ℓ_1) for vector-valued image restoration in this chapter. By introducing some auxiliary variables and Lagrange multipliers, we develop a fast algorithm based on alternating direction method of multipliers (ADMM) for the proposed model, which allows the use of the fast Fourier transform (FFT), generalized soft threshold formulas and projection method. Extensive experiments have been conducted on both synthetic and real color images, which validate that the proposed approach performs better in image restoration, and show advantages of the proposed ADMM over algorithms based on traditional gradient descent method (GDM) in terms of computational efficiency.

Next we model illusory contour reconstruction as a minimization problem with a tractable variational level set formulation, utilizing Eulers elastica to reconstruct the illusory boundaries. However, this kind of formulation is very difficult to solve numerically as it is hard to implement such optimization algorithms efficiently in practice. In this chapter, we propose an reduced variational level set formulation by taking the level set functions as signed distance functions. Technically, an alternating direction method of multipliers incorporated with the projection method (ADMM-P) is developed by introducing some auxiliary variables, Lagrange multipliers and applying an alternating optimization strategy. The corresponding level set functions are regarded as signed distance functions during the computation process using a simple algebraic projection method, which avoids the traditional re-initialization process for conventional level set functions. Extensive experiments have been conducted on both synthetic and real images, which validated the proposed approach, and demonstrated the advantages of the proposed ADMM-P method over the existing algorithms based on traditional gradient descent method (GDM) in terms of computational efficiency.

Subsequently, a new variational model is proposed for image segmentation based on active contours, nonlinear diffusion and level sets. It includes a Chan-Vese model-based data fitting term and a regularized term that uses the potential functions (PF) of nonlinear diffusion. The first term can segment the image by region partition instead of having to rely on edge information. The second term is capable of preserving image edges as well as smoothing noisy regions. To improve the computational efficiency, the implementation of the proposed model does not directly solve the high order nonlinear partial differential equations but instead exploit the efficient ADMM. In particular, we propose a new fast algorithm, normal vector projection method (NVPM), based on alternating optimization method and normal vector projection. Its stability can be the same as ADMM and it converges faster. Extensive numerical experiments on grey and colour images validate the effectiveness of the proposed model and the efficiency of the algorithms. In the next chapter, the possibility of introducing excellent theories or mechanisms into variational methods to tackle some existing restrictions will be investigated, which is also the second focus of this thesis.

Chapter 4

Stochastic Programming Based Variational Segmentation Approach for Noisy Images

4.1 Introduction

In image segmentation, the Mumford-Shah model [116] is regarded as one of the most significant region-based models and has been applied to many applications. In 2001, the two-phase Chan-Vese (CV) model [40] was proposed to detect objects in a given image. With the increasing complexity of images, the multiphase segmentation models [69, 94] were proposed and these models mainly represent different regions by using the level set functions [164, 166, 176, 210]. In order to reduce the number of level set functions, Chan et al. proposed a multiphase segmentation model [171], which is a generalization of CV model.

Some specific segmentation models [108, 127, 146] were also established subsequently according to different noise distribution. They obtained the characteristic information contained in images by estimating corresponding parameters. When dealing with noisy images, it is known that many segmentation problems need a suitable noise model, e.g., synthetic aperture radar, positron emission tomography, electron micrograph or medical ultrasound imaging, etc. Especially when the data were collected with poor statistics, it is necessary to

⁰© 2020 Springer. This chapter is based on, as well as with permission from [Tan, L., Li, L., Liu, W., Sun, J., & Zhang, M. A Novel Euler's Elastica-Based Segmentation Approach for Noisy Images Using the Progressive Hedging Algorithm. *Journal of Mathematical Imaging and Vision*, 2020. DOI: <https://doi.org/10.1007/s10851-019-00920-0>].

consider the influence of the noise probability distribution in segmentation implementation.

Recently, authors in [159, 166, 212] made some progress in achieving illusory contour recovery while doing segmentation, which can identify absent boundaries or missing shapes successfully without necessary region features. In detail, the fitting terms of two-phase CV model and the Euler's elastica term [159] were employed as the regularization in [212]. Its major contribution was that the missing boundaries were interpolated automatically without specifying the regions. The work [166] improved the segmentation with depth problem [121, 210] and achieved acceleration via the strategies of model simplification and constraint projection. Its significant performance enhancements included shape reconstruction of occluded objects and determination of their ordering relation in a specific scene based on only one single image. Many other works illustrated that the curvature-related terms have played crucial roles in the boundary reconstruction [83, 164] and image restoration [163, 165, 211] with the capacity of producing excellent edge and corner preservation results. All of these researches show the significant potential for curvature-based methods.

However, the segmentation models mentioned above cannot be directly applied for noisy images when the type of noise is unknown or there are more than one type of noise in the image. The reason is that in these models there exists a one-to-one mapping between the parameters to be evaluated and the noisy images with some probability density distribution. Besides, the curvature-related terms will bring extra computational complexity due to the existence of nonlinear higher-order derivatives. This issue also appears in other variational models such as the non-texture image inpainting [191] and image denoising [186] with features (edge, corner, smoothness, contrast, etc.) preservation. Hence it is essential to take some mathematical optimization techniques, for example with global solution, stability guarantee and calculation acceleration, into consideration in the process of algorithm design.

To deal with stochastic noises in image segmentation in general, [134, 135] have given us a great source of inspiration from the perspective of theory analysis. The authors there extended the progressive hedging algorithm (PHA) for multistage stochastic variational inequality problems and solved the stochastic complementarity problems in a two-stage formulation. One of our major motivations in this chapter is to embed stochastic property into segmentation energy functional for images with unknown noises or arbitrary damages and solve the functional using PHA. Furthermore, Euler's elastica term will be considered

as the regularization in our variational formulations since it has better properties in dealing with image feature information. Last but not least, in order to improve the computational efficiency and solve problems caused by the non-convex, non-differentiable, nonlinear and higher-order terms involved in Euler’s elastica-related functional, algorithms such as alternating direction method of multipliers (ADMM) [61, 186, 191], curvature-weighted approach [11, 50, 191] and convex relaxation method [32, 36] will be systematically designed as a fusion for energy minimization. In this chapter, the performance of the proposed approach in terms of accuracy, stability and practicability is validated through numerical experiments.

Our contributions in this chapter can be summarized in the following aspects:

- We propose novel formulations for image segmentation with stochastic noises by transforming the original energy minimization problems into the optimization framework of stochastic programming.
- Euler’s elastica term is employed to achieve meaningful completion of missing boundaries, reconstruction of occluded structures and segmentation simultaneously, which further enhances the performance of image segmentation .
- Our novel variational formulations are applied for the two-phase Euler’s elastica based segmentation and segmentation with depth in both grayscale and color spaces respectively.
- A general numerical algorithm based on PHA is proposed. Fusion of ADMM and curvature weighted approach (ADMM-C) is designed for the minimization of the convex-relaxed variational energy related subproblems. The minimization problems derived from ADMM-C will be efficiently solved by Fast Fourier transform (FFT) [50, 114] and analytical soft threshold formulas [164, 165, 179].

The rest of this chapter is structured as follows. Our proposed approach and algorithm framework are illustrated in Section 4.2. The experiments conducted with performance evaluation and comparison are described in Section 4.3 followed by the summary in Section 4.4.

4.2 Novel Formulations for Different Segmentation Problems Incorporating Stochastic Noises Via Progressive Hedging

The focus of our work is to formulate the noisy image segmentation process as a two-stage stochastic programming problem, in which the first-stage decision is to determine the curve of the object in the image and the randomness ξ is the distributions of noise, then we adopt PHA framework to handle the curve under stochastic noises, which will be described in detail in this section. Motivated by the research using PHA [134] to solve the minimization problem of stochastic programming, we aim to propose novel formulations tackling different segmentation issues in consideration of the advantages of Euler's elastica term and the influence of unknown noises. In this way, we not only can fulfill general segmentation tasks with the classic model, but also can deal with some bad situations such as low quality images with large noises, absent boundaries, missing shapes or occlusion. Then our proposed formulations are incorporated with the developed ADMM-C algorithm to obtain the optimal solutions efficiently.

Firstly, we attempt to formulate a new problem and apply progressive hedging algorithm in stochastic programming to improve the segmentation performance. The obvious advantage of this approach is not just that it is robust to noise, it can also deal with stochastic noises, which means we do not need to know what specific types of noises are contained in the image. Fig. 4-1 presents the comparison of results to further demonstrate this feature. The original image is given in (a). (b) presents the image with unknown noises. Results without or with PHA in stochastic programming are shown in (c) and (d) respectively. Here we use the classic CV model [40] for segmentation. This application can also be utilized as a preprocessing procedure providing the initialization of binary level set functions for segmentation with depth. Specifically, the model can be expressed as the minimization of the following energy functional

$$\begin{aligned}
 E_{\text{SP-gray}}^{\text{CV}}(\theta_\xi, \phi_\xi) &= \alpha_1 \int_{\Omega} Q_1(x, \theta_1(\xi)) \phi(\xi) dx + \alpha_2 \int_{\Omega} Q_2(x, \theta_2(\xi)) (1 - \phi(\xi)) dx \\
 &+ \gamma \int_{\Omega} |\nabla \phi(\xi)| dx + \int_{\Omega} (v^k(\xi) \phi(\xi) + \frac{\tau}{2} (\phi(\xi) - \phi_\xi^k)^2) dx, \quad (4.1)
 \end{aligned}$$

which is an PHA-extended variational model from model (2.30). Here each element ξ_i of set $\xi = (\xi_1, \xi_2, \dots, \xi_N)$ represents one specific noise type and $Q(x, \theta(\xi_i))$ is the potential

function estimated according to its noise distribution, which are all summarized in Table 2.2. Analog to the PHA framework in two-stage stochastic programming introduced in Section 4.2.1, binary level set functions $\phi(\xi) = (\phi(\xi_1), \phi(\xi_2), \dots, \phi(\xi_N))$ are the variables of Stage 1, which should be obtained separately by solving their minimization problems. Likewise, ϕ_ξ belongs to the variable of Stage 2 relying on the projection mappings $P_{\mathcal{N}}$ stated in (2.36) implemented on the obtained variables $(\phi(\xi_1), \phi(\xi_2), \dots, \phi(\xi_N))$ from Stage 1. Before starting Stage 1 again, all the variables $(\phi(\xi_1), \phi(\xi_2), \dots, \phi(\xi_N))$ are updated with ϕ_ξ . It is hence that ϕ_ξ can be achieved by considering stochastic noise information through $P_{\mathcal{N}}$. For the last two terms derived from PHA that are detailed in **Algorithm 2.1** of Chapter 2, here w -term is replaced with v -term to avoid confusion since \vec{w} will be introduced as auxiliary variables in our algorithm design. v -term also need to be updated after ϕ_ξ is obtained. Then r -term is changed into τ -term accordingly. The necessity of these two terms is also discussed in this chapter (Section 4.3.1). Until now, stochastic property of noises has been successfully formulated into a classic CV segmentation model by taking advantage of PHA framework.

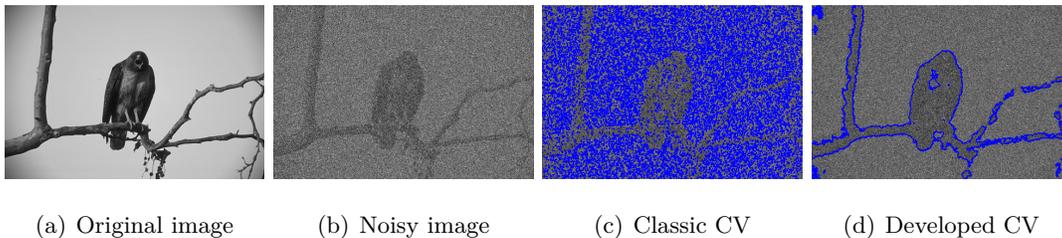


Figure 4-1: Example of PHA in stochastic programming for segmentation improvement. (a): original image; (b): noisy image; (c): results obtained by CV model; (d): results obtained by CV model with PHA in stochastic programming.

The region-based CV model is well known for its capabilities of being insensitive to noises and capturing desired results even on weak boundaries. From Fig. 4-1(c), we can see that the performance of CV model may be influenced by stronger noises. Fig. 4-1(d) validates that the classic CV model using PHA in stochastic programming can achieve improvements for standard segmentation.

This improved CV-PHA model can be extended into Euler's elastica based segmentation functional. The new models with Euler's elastica term possess the following abilities: 1) producing standard segmentation results as good as the classic CV model [40]; 2) achieving

better performance on images with strong unknown noises, missing boundaries or fuzzy areas (e.g. it can complete the disconnected parts of the branch in Fig. 4-1(d), while CV-PHA functional (4.1) without Euler’s elastica term is not able to); and 3) additionally, recovering occluded shapes while multiphase segmentation models [69, 94, 171] without Euler’s elastica term can not.

4.2.1 Two-phase Segmentation Based Application

Based on the two-phase Euler’s elastica based segmentation formulated in (2.15) and the CV-PHA model (4.1), we propose a novel segmentation model dealing with different noises expressed as the following stochastic programming (SP) form. The reconstructed contour can be obtained by minimizing the following energy functional with respect to $\phi(\xi)$.

Model I: Two-phase Euler’s elastica based Segmentation in Stochastic Programming

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-gray}}^{\text{Euler's elastica}}(\theta_\xi, \phi_\xi) \right. \\
= & \alpha_1 \int_{\Omega} Q_1(x, \theta_1(\xi)) \phi(\xi) dx + \alpha_2 \int_{\Omega} Q_2(x, \theta_2(\xi)) (1 - \phi(\xi)) dx \\
& \left. + \int_{\Omega} \left(\alpha + \beta \left| \nabla \cdot \frac{\nabla \phi(\xi)}{|\nabla \phi(\xi)|} \right| \right) |\nabla \phi(\xi)| dx + \int_{\Omega} (v^k(\xi) \phi(\xi) + \frac{\tau}{2} (\phi(\xi) - \phi_\xi^k)^2) dx \right\} \quad (4.2)
\end{aligned}$$

$\xi = (\xi_1, \xi_2, \dots, \xi_N)$ represent different noise distributions and $Q_i(x, \theta_i(x))$ contain the stochastic information need to be estimated. $\phi(\xi)$ is the optimal solution of (4.2) under distribution ξ . Here we continue the definition of $\phi(\xi)$ in [212] using the binary representation which can also be explained as a substitution $\phi = H(\varphi)$. As described in [212], this binary representation was originally used for finding the global minimizer, and it can also reduce the computational complexity to some extent such as avoiding calculations associated with level sets. It can be observed that (4.2) will reduce to the CV-PHA model (shown as Functional (4.1)) when β equals 0. In this way, we can also regard CV based models as particular cases of our proposed segmentation models. Functional (4.1) gives a good example that incorporating different noise distributions into one formulation based on PHA can enhance the ability of handling strong noise better than applying them separately and the improved performance has been validated in Figure 4-1. In addition, Euler’s elastica

was studied for visual construction in [83, 121, 210], of which the major advantages lied in the effect of reconstructing illusory contours or recovering occluded shapes. From [164, 166], it was found that the good property of Euler's elastica term is also reflected in its robustness against noises. According to the limitation of functional (4.1) discussed previously, we aim to employ Euler's elastica term as the regularization in the PHA-based segmentation framework so as to make progress in performance for image segmentation.

According to the segmentation model for vector-valued images proposed in [37], the averages of the data terms over all channels are used for coupling. Let $f = (f_1, f_2, \dots, f_m)$ be a original color image defined on a domain Ω . Then function Q should also be in the multichannel form (Q_1, Q_2, \dots, Q_m) . In fact, our proposed model used to solve color image segmentation can be stated as follows:

Model II: Two-phase Euler's elastica based Segmentation in Stochastic Programming (Multi-Channel)

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-color}}^{\text{Euler's elastica}}(\theta_\xi, \phi_\xi) \right. \\
&= \alpha_1 \int_{\Omega} \sum_{l=1}^m Q_{1l}(x, \theta_{1l}(\xi)) \phi(\xi) dx + \alpha_2 \int_{\Omega} \sum_{l=1}^m Q_{2l}(x, \theta_{2l}(\xi)) (1 - \phi(\xi)) dx \quad (4.3) \\
& \left. + \int_{\Omega} (\alpha + \beta \left| \nabla \cdot \frac{\nabla \phi(\xi)}{|\nabla \phi(\xi)|} \right|) |\nabla \phi(\xi)| dx + \int_{\Omega} (v^k(\xi) \phi(\xi) + \frac{\tau}{2} (\phi(\xi) - \phi_\xi^k)^2) dx \right\}
\end{aligned}$$

where $l = 1, 2, \dots, m$ denotes the number of layers of a vector-valued image. Hence novel Euler's elastica based formulations embedding stochastic noises for two-phase segmentation are obtained. In the following section we shall describe the numerical procedure of the proposed models for solving the multistage stochastic programming problem.

4.2.2 PHA with ADMM-C Algorithm for Two-phase Segmentation Application

In order to demonstrate the precise numerical procedure of PHA in solving (4.2) and (4.3), we focus on the general model suitable for both gray space and color space cases. Detailed proofs for convergence of this algorithm are provided in [134]. The original minimization

problems (4.2) and (4.3) are based on separate sub optimization problems,

$$\arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \{E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_\xi, \phi_\xi)\} \Rightarrow \begin{cases} \arg \min_{\theta, \phi \in \{0,1\}} \{E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_1}, \phi_{\xi_1})\} \\ \arg \min_{\theta, \phi \in \{0,1\}} \{E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_2}, \phi_{\xi_2})\} \\ \arg \min_{\theta, \phi \in \{0,1\}} \{E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_3}, \phi_{\xi_3})\} \\ \arg \min_{\theta, \phi \in \{0,1\}} \{E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_4}, \phi_{\xi_4})\} \end{cases} \quad (4.4)$$

where $\xi = (\xi_1, \xi_2, \dots, \xi_N)$ refer to noise distributions. For each distribution ξ_i , it has a known probability $p(\xi_i) > 0$ which can be set empirically through experiments, and the sum $\sum_i p(\xi_i) = 1$ has to be guaranteed. According to PHA approach, ϕ_ξ can be obtained by the following steps.

i). the minimization problems need to be solved separately, which are given in the right side of (4.4).

$$\begin{aligned} (\theta_{\xi_i}^{k+1}, \phi_{\xi_i}^{k+1}) &= \arg \min_{\theta, \phi \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_i}, \phi_{\xi_i}) \right. \\ &= \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1(\xi_i)) \phi(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2(\xi_i)) (1 - \phi(\xi_i)) dx \\ &\left. + \int_{\Omega} (\alpha + \beta |\kappa(\xi_i)|) |\nabla \phi(\xi_i)| dx + \int_{\Omega} (v^k(\xi_i) \phi(\xi_i) + \frac{\tau}{2} (\phi(\xi_i) - \phi_\xi^k)^2) dx \right\} \end{aligned} \quad (4.5)$$

where $\kappa = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}$. In the gray space, there is only one layer of the image information to be calculated, which means $Q^*(x, \theta(\xi_i)) = Q(x, \theta(\xi_i))$. In the color space, $Q^*(x, \theta(\xi_i))$ should be substituted with the coupling terms $\sum_l Q_l(x, \theta_l(x_i))$. $Q^*(x, \theta(\xi_i))$ is the potential function for specific noise distribution in each channel of the image. Table 2.2 shows the representations of $Q(x, \theta(\xi_i))$ and estimations of parameters $\theta(\xi_i)$. ϕ_{ξ_i} is the optimal solution of (4.5) in distribution ξ_i .

ii). Next, all the obtained optimal solutions are utilized to gain the final optimum ϕ_ξ .

$$\phi_\xi^{k+1} = \sum_{i=1}^N p(\xi_i) \phi_{\xi_i}^{k+1}, \quad (4.6)$$

iii). At last, the sub problems' solutions $\phi^k(\xi_i)$ and the Lagrangian multipliers $v^k(\xi_i)$

need to be updated at the end of each iteration.

$$\begin{cases} \phi^{k+1}(\xi_1) = \phi_\xi^{k+1} \\ \phi^{k+1}(\xi_2) = \phi_\xi^{k+1} \\ \vdots \\ \phi^{k+1}(\xi_N) = \phi_\xi^{k+1} \end{cases}, \begin{cases} v^{k+1}(\xi_1) = v^k(\xi_1) + \tau(\phi^{k+1}(\xi_1) - \phi_\xi^{k+1}) \\ v^{k+1}(\xi_2) = v^k(\xi_2) + \tau(\phi^{k+1}(\xi_2) - \phi_\xi^{k+1}) \\ \vdots \\ v^{k+1}(\xi_N) = v^k(\xi_N) + \tau(\phi^{k+1}(\xi_N) - \phi_\xi^{k+1}) \end{cases}. \quad (4.7)$$

The updated $\phi^{k+1}(\xi_i)$, $v^{k+1}(\xi_i)$ and the parameters $\theta^{k+1}(\xi_i)$ derived from (4.5) are passed to the next iteration from step **i**).

To solve the minimization problems (4.5) separately, both the simplification and effectiveness of the algorithm should be considered. There are three main computational difficulties in the functional as listed below, followed by the corresponding algorithm design.

- (i) One main challenge of the Euler's elastica based functional is due to the non-smoothness and non-convexity of $g(\kappa) = \alpha + \beta|\kappa|$. As described in [11, 50, 191], it is more efficient when the curvature term $g(\kappa)$ is computed separately from the functional (4.5). Inspired by the concept of curvature weighted approach, we can rewrite functional (4.5) as the following simplified version

$$\begin{aligned} (\theta_{\xi_i}^{k+1}, \phi_{\xi_i}^{k+1}) &= \arg \min_{\theta, \phi \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_i}, \phi_{\xi_i}) \right. \\ &= \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1(\xi_i)) \phi(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2(\xi_i)) (1 - \phi(\xi_i)) dx \\ &\quad \left. + \int_{\Omega} g(\kappa(\xi_i)) |\nabla \phi(\xi_i)| dx + \int_{\Omega} (v^k(\xi_i) \phi(\xi_i) + \frac{\tau}{2} (\phi(\xi_i) - \phi_\xi^k)^2) dx \right\} \end{aligned} \quad (4.8)$$

The proposed approach essentially reduces the minimization problem (4.8) to a total variation type [140]. Here division by zero in $g(\kappa)$ should be avoided. In practice, the denominator is often replaced by $|\nabla \phi|_\epsilon = \max(\epsilon, |\nabla \phi|)$ (ϵ is an arbitrarily small positive parameter). Then $g(\kappa(\xi_i))$ is represented as $\nabla \cdot (\nabla \phi(\xi_i)) / |\nabla \phi(\xi_i)|_\epsilon$.

- (ii) Note that the binary constraint for ϕ will also cause non-convexity in subproblems (4.8). As demonstrated in [36], certain non-convex minimization problems can be

equivalent to the following convex minimization problems

$$\begin{aligned}
(\theta_{\xi_i}^{k+1}, \phi_{\xi_i}^{k+1}) &= \arg \min_{\theta, \phi \in [0,1]} \left\{ E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_i}, \phi_{\xi_i}) \right. & (4.9) \\
&= \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1(\xi_i)) \phi(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2(\xi_i)) (1 - \phi(\xi_i)) dx \\
&\quad \left. + \int_{\Omega} g(\kappa(\xi_i)) |\nabla \phi(\xi_i)| dx + \int_{\Omega} (v^k(\xi_i) \phi(\xi_i) + \frac{\tau}{2} (\phi(\xi_i) - \phi_{\xi}^k)^2) dx \right\}
\end{aligned}$$

This convex minimization scheme could find global minimizers for (4.8) by thresholding the solution of (4.9), which was classified as a continuous min-cut algorithm. Together with its equivalent form, the continuous max-flow algorithm, the min-cut algorithm has been proved to be an exact convex relaxation of the original problem as discussed in [13, 197].

- (iii) Another critical issue for solving (4.9) is the inevitable high order derivatives in numerical implementation, which is tedious and prone to errors. A developed ADMM-C algorithm is designed to apply on each sub problem by introducing auxiliary variables, Lagrangian multipliers and an alternating directional optimization strategy.

The detailed implementation on solving each sub-problem by the ADMM-C algorithm is presented as below. Firstly, some auxiliary variables are introduced, i.e., $\vec{w}(\xi_i) = [w_1(\xi_i), w_2(\xi_i)]^T$ with property $\vec{w} \approx \nabla \phi(\xi_i)$ and the Lagrangian multipliers $\vec{\lambda}(\xi_i) = [\lambda_1(\xi_i), \lambda_2(\xi_i)]^T$. (4.9) can then be transformed into the following augmented Lagrangian functional

$$\begin{aligned}
(\theta_{\xi_i}^{k+1}, \phi_{\xi_i}^{k+1}, \vec{w}_{\xi_i}^{k+1}) &= \arg \min_{\theta, \phi \in [0,1]} \left\{ E_{\text{SP-general}}^{\text{Euler's elastica}}(\theta_{\xi_i}, \phi_{\xi_i}) \right. & (4.10) \\
&= \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1(\xi_i)) \phi(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2(\xi_i)) (1 - \phi(\xi_i)) dx \\
&\quad + \int_{\Omega} g(\kappa(\xi_i)) |\vec{w}(\xi_i)| dx + \int_{\Omega} (v^k(\xi_i) \phi(\xi_i) + \frac{\tau}{2} (\phi(\xi_i) - \phi_{\xi}^k)^2) dx \\
&\quad \left. + \int_{\Omega} (\vec{\lambda}(\xi_i) \cdot (\vec{w}(\xi_i) - \nabla \phi(\xi_i)) + \frac{\mu}{2} (\vec{w}(\xi_i) - \nabla \phi(\xi_i))^2) dx \right\}
\end{aligned}$$

where μ is a positive penalty parameter. It is worth noting that this kind of simple structure of (4.10) requires fewer variables and parameters compared with other works [159, 165] using the ADMM to deal with the curvature term directly. After the initialization of $\phi^0(\xi_i)$, $\vec{w}^0(\xi_i)$ and $\vec{\lambda}^0(\xi_i)$, a minimization problem is carried out in each step with respect to one variable while keeping other variables fixed temporarily. When the alternating optimization for all

the variables is completed, the Lagrangian multipliers will be updated subsequently. This gives

$$\theta_{\xi_i}^{k+1} = \arg \min_{\theta} \left\{ \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1(\xi_i)) \phi^k(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2(\xi_i)) (1 - \phi^k(\xi_i)) dx \right\}, \quad (4.11)$$

$$\begin{aligned} \phi_{\xi_i}^{k+1} &= \arg \min_{\phi \in [0,1]} \left\{ \alpha_1 \int_{\Omega} Q_1^*(x, \theta_1^{k+1}(\xi_i)) \phi(\xi_i) dx + \alpha_2 \int_{\Omega} Q_2^*(x, \theta_2^{k+1}(\xi_i)) (1 - \phi(\xi_i)) dx \right. \\ &+ \int_{\Omega} (v^k(\xi_i) \phi(\xi_i) + \frac{\tau}{2} (\phi(\xi_i) - \phi_{\xi}^k)^2) dx \\ &+ \left. \int_{\Omega} (\vec{\lambda}^k(\xi_i) \cdot (\vec{w}^k(\xi_i) - \nabla \phi(\xi_i)) + \frac{\mu}{2} (\vec{w}^k(\xi_i) - \nabla \phi(\xi_i))^2) dx \right\} \end{aligned} \quad (4.12)$$

$$\begin{aligned} \vec{w}_{\xi_i}^{k+1} &= \arg \min_{\vec{w}} \left\{ \int_{\Omega} g(\kappa^{k+1}(\xi_i)) |\vec{w}(\xi_i)| dx + \int_{\Omega} \vec{\lambda}^k(\xi_i) \cdot (\vec{w}(\xi_i) - \nabla \phi^{k+1}(\xi_i)) \right. \\ &+ \left. \frac{\mu}{2} (\vec{w}(\xi_i) - \nabla \phi^{k+1}(\xi_i))^2 dx \right\}, \quad \text{where } g(\kappa^{k+1}(\xi_i)) = \nabla \cdot \frac{\nabla \phi_{\xi_i}^{k+1}}{|\nabla \phi_{\xi_i}^{k+1}|_{\epsilon}}, \end{aligned} \quad (4.13)$$

$$\vec{\lambda}_{\xi_i}^{k+1} = \vec{\lambda}_{\xi_i}^k + \mu (\vec{w}^{k+1}(\xi_i) - \nabla \phi^{k+1}(\xi_i)). \quad (4.14)$$

To obtain $\theta^{k+1} = (\mu, \sigma)$: In the $k + 1$ step of the proposed ADMM-C, the average image intensity values μ_{ξ_i} as well as variances σ_{ξ_i} in the foreground and background can be obtained by using the standard variational method for (4.11). Table 2.2 gives all the solutions for distributions $\xi = (\xi_1, \xi_2, \dots, \xi_N)$.

To obtain ϕ^{k+1} : Optimal value of ϕ_{ξ_i} for a certain distribution ξ_i is obtained by solving the minimization of (4.12) with respect to $\phi(\xi_i)$. We can get the update rule based on the corresponding Euler-Lagrange equations

$$(-\mu \Delta + \tau) \phi(\xi_i) = \tau \phi_{\xi}^k - r_{\xi_i}(\theta_1^{k+1}, \theta_2^{k+1}) - v^k(\xi_i) - \nabla \cdot \vec{\lambda}^k(\xi_i) - \mu \nabla \cdot \vec{w}^k(\xi_i), \quad (4.15)$$

where $r_{\xi_i}(\theta_1^{k+1}, \theta_2^{k+1}) = \alpha_1 Q_1^*(x, \theta_1^{k+1}(\xi_i)) - \alpha_2 Q_2^*(x, \theta_2^{k+1}(\xi_i))$. As discussed in [114], equation (4.15) is a screened Poisson equation for which the Fast Fourier transform (FFT) [50, 114] is a well-known solver with very low computational cost for imaging problems. Here FFT is applied to further improve the calculation efficiency. Equation (4.15) can be

rewritten as

$$F^* L F \phi(\xi_i) = \tau \phi_\xi^k - r_{\xi_i}(\theta_1^{k+1}, \theta_2^{k+1}) - v^k(\xi_i) - \nabla \cdot \vec{\lambda}^k(\xi_i) - \mu \nabla \cdot \vec{w}^k(\xi_i), \quad (4.16)$$

where $L = -\mu F \Delta F^* + \tau$ and F^* is the discrete inverse Fourier transform. $\phi_{\xi_i}^{k+1}$ can then be obtained as follows

$$\phi_{\xi_i}^{k+1} = F^*(L^{-1} F(\tau \phi_\xi^k - r_{\xi_i}(\theta_1^{k+1}, \theta_2^{k+1}) - v^k(\xi_i) - \nabla \cdot \vec{\lambda}^k(\xi_i) - \mu \nabla \cdot \vec{w}^k(\xi_i))). \quad (4.17)$$

To obtain \vec{w}^{k+1} : The minimization problem (4.13) of \vec{w} can be solved via the generalized soft thresholding formula [164, 165, 179], which is given by

$$\vec{w}_{\xi_i}^{k+1} = \max \left(\left| \nabla \phi_{\xi_i}^{k+1} - \frac{\vec{\lambda}^k(\xi_i)}{\mu} \right| - \frac{g(\kappa^{k+1}(\xi_i))}{\mu}, 0 \right) \frac{\nabla \phi_{\xi_i}^{k+1} - \frac{\vec{\lambda}^k(\xi_i)}{\mu}}{\left| \nabla \phi_{\xi_i}^{k+1} - \frac{\vec{\lambda}^k(\xi_i)}{\mu} \right|}. \quad (4.18)$$

The overall algorithm for the two-phase Euler's elastica based segmentation in stochastic programming is presented in a pseudo code format as follows.

Algorithm 4.1 Computing framework for (4.2) and (4.3) via PHA

Input: $\phi^0(\xi_i), p(\xi_i), v^0(\xi_i), \alpha, \beta, \tau, \alpha_1, \alpha_2$

for $k \geq 1$, do the following steps recurrently

- 1: Obtain $\phi_{\xi_i}^{k+1}$ via **Algorithm 4.2**
- 2: Update ϕ_ξ^{k+1} via Equation (4.6)
- 3: Update $\phi^{k+1}(\xi_i), v^{k+1}(\xi_i)$ via Equation (4.7)
- 4: **if** some stopping criteria (given in Section 4.3.5) are satisfied **break**

Return optimal value ϕ_ξ^{k+1} after thresholding

Algorithm 4.2 Details for step 1 in Algorithm 4.1 via ADMM-C

If $k = 1$

input $\vec{w}^0(\xi_i), \vec{\lambda}^0(\xi_i), \mu$

else solve the following problems alternatively

- 1: Update $\theta_{\xi_i}^{k+1}$ according to distribution laws
 - 2: Update ϕ_ξ^{k+1} via minimization problem (4.12)
 - 3: Update $\vec{w}_{\xi_i}^{k+1}$ via minimization problem (4.13)
 - 4: Update $\vec{\lambda}_{\xi_i}^{k+1}$ via (4.14) using gradient ascent method
-

The idea of the ADMM-C method is intentionally applied for the non-convex, non-smooth and non-linear problems in Euler's elastics and it has attracted extensive attention. ADMM algorithm has been analytically studied in [72, 99] and it has been used in many other applications [117, 190] to successfully achieve excellent performances via solving a variety of non-convex problems. The similar idea of ADMM-C in [50, 191] is adopted here to deal with the sub-problems derived from PHA framework.

4.2.3 Segmentation with Depth Based Application

We intend to use stochastic programming skills similar to the ones applied in the two-phase issue. Based on the original segmentation with depth model for gray space in (2.16), the energy functional can be established by introducing random noise set as follows

Model III: Segmentation with Depth in Stochastic Programming

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-gray}}^{\text{Depth}}(\theta_\xi, \phi_\xi) \right. & (4.19) \\
& = \sum_{h=1}^n \int_{\Omega} (\alpha + \beta |\nabla \cdot \frac{\nabla \phi_h(\xi)}{|\nabla \phi_h(\xi)}|) |\nabla \phi_h(\xi)| dx + \sum_{h=1}^n \int_{\Omega} Q_h(x, \theta_h(\xi)) \phi_h(\xi) \prod_{j=1}^{h-1} (1 - \phi_j(\xi)) dx \\
& \left. + \int_{\Omega} Q_{n+1}(x, \theta_{n+1}(\xi)) \prod_{j=1}^n (1 - \phi_j(\xi)) dx + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi) \cdot \phi_h(\xi) + \frac{\tau}{2} (\phi_h(\xi) - \phi_h^k(\xi))^2) dx \right\}
\end{aligned}$$

where $h = 1, 2, \dots, n$ denotes the number of objects in the image, and $\phi_{n+1}(\xi) = 1$ is set for consistency of description. The above segmentation with depth incorporating stochastic noises model can be extended to multichannel case. Analog to the coupling approach used in (4.3), the formulation can be written as

Model IV: Segmentation with Depth in Stochastic Programming (Multi-Channel)

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-color}}^{\text{Depth}}(\theta_\xi, \phi_\xi) \right. & (4.20) \\
& = \sum_{h=1}^n \int_{\Omega} (\alpha + \beta |\nabla \cdot \frac{\nabla \phi_h(\xi)}{|\nabla \phi_h(\xi)|}|) |\nabla \phi_h(\xi)| dx \\
& + \sum_{h=1}^n \int_{\Omega} \sum_{l=1}^m Q_{hl}(x, \theta_{hl}(\xi)) \phi_h(\xi) \prod_{j=1}^{h-1} (1 - \phi_j(\xi)) dx \\
& + \int_{\Omega} \sum_{l=1}^m Q_{(n+1)l}(x, \theta_{(n+1)l}(\xi)) \prod_{j=1}^n (1 - \phi_j(\xi)) dx \\
& \left. + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi) \cdot \phi_h(\xi) + \frac{\tau}{2} (\phi_h(\xi) - \phi_{h(\xi)}^k)^2) dx \right\}
\end{aligned}$$

According to the curvature-weighted approach used in (4.8), the simplified version can be directly written as

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Depth}}(\theta_\xi, \phi_\xi) \right. & (4.21) \\
& = \sum_{h=1}^n \int_{\Omega} g(\kappa_h(\xi)) |\nabla \phi_h(\xi)| dx + \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h(\xi)) \chi_h(\xi) dx \\
& \left. + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}(\xi)) \chi_{n+1}(\xi) dx + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi) \cdot \phi_h(\xi) + \frac{\tau}{2} (\phi_h(\xi) - \phi_{h(\xi)}^k)^2) dx \right\}
\end{aligned}$$

where $g(\kappa_h(\xi)) = \nabla \cdot (\nabla \phi_h(\xi)) / |\nabla \phi_h(\xi)|_\epsilon$ and the definition of $|\nabla \phi_h(\xi)|_\epsilon$ is given in (4.8). $Q_h^*(x, \theta_h(\xi))$ is $Q_h(x, \theta_h(\xi))$ for gray space issue and $\sum_l Q_{hl}(x, \theta_{hl}(\xi))$ for color space issue. The characteristic function for the h -th region reads $\chi_h(\xi) = \phi_h(x) \prod_{j=1:h-1} (1 - \phi_j(\xi))$. Particularly, $\chi_{n+1}(\xi) = \phi_{n+1}(\xi) \prod_{j=1:n} (1 - \phi_j(\xi)) = \prod_{j=1:n} (1 - \phi_j(\xi))$ representing the $(n+1)$ -th region background. In the next section we shall solve the above general formulation under PHA with detailed implementation.

4.2.4 PHA with ADMM-C Algorithm for Segmentation with Depth

The numerical solution is similar to those used in the two-phase imaging tasks. The main program loop for PHA is shown as follows

i). the sub minimization problems of (4.21) need to be solved separately, which gives

$$\begin{aligned}
& (\theta_h^{k+1}, \phi_h^{k+1}) \\
= & \arg \min_{\theta_h, \phi_h \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Depth}}(\theta_h(\xi_i), \phi_h(\xi_i)) \right. \\
= & \sum_{h=1}^n \int_{\Omega} g(\kappa_h(\xi_i)) |\nabla \phi_h(\xi_i)| dx + \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h(\xi_i)) \chi_h(\xi_i) dx \\
& \left. + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}(\xi_i)) \chi_{n+1}(\xi_i) dx + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi_i) \cdot \phi_h(\xi_i) + \frac{\tau}{2} (\phi_h(\xi_i) - \phi_h^k(\xi_i))^2) dx \right\}
\end{aligned} \tag{4.22}$$

ii). all the obtained optimal solutions $(\phi_h(\xi_1), \phi_h(\xi_2), \dots, \phi_h(\xi_N))$ are utilized to gain the final optimum $\phi_h(\xi)$.

$$\phi_h^{k+1}(\xi) = \sum_{i=1}^N p(\xi_i) \phi_h^{k+1}(\xi_i). \tag{4.23}$$

iii). the sub problems' solutions $\phi_h^k(\xi_i)$ and the Lagrangian multipliers $v_h^k(\xi_i)$ need to be updated at the end of each iteration.

$$\left\{ \begin{array}{l} \phi_h^{k+1}(\xi_1) = \phi_h^{k+1}(\xi) \\ \phi_h^{k+1}(\xi_2) = \phi_h^{k+1}(\xi) \\ \vdots \\ \phi_h^{k+1}(\xi_N) = \phi_h^{k+1}(\xi) \end{array} \right\}, \left\{ \begin{array}{l} v_h^{k+1}(\xi_1) = v_h^k(\xi_1) + \tau(\phi_h^{k+1}(\xi_1) - \phi_h^k(\xi_1)) \\ v_h^{k+1}(\xi_2) = v_h^k(\xi_2) + \tau(\phi_h^{k+1}(\xi_2) - \phi_h^k(\xi_2)) \\ \vdots \\ v_h^{k+1}(\xi_N) = v_h^k(\xi_N) + \tau(\phi_h^{k+1}(\xi_N) - \phi_h^k(\xi_N)) \end{array} \right\}. \tag{4.24}$$

updated $\phi_h^{k+1}(\xi_i)$, $v_h^{k+1}(\xi_i)$ and the parameters $\theta^{k+1}(\xi_i)$ derived from (4.22) are passed to the next iteration from step **i**).

In order to solve the minimization problems (4.22) in step **i**) separately, the proposed ADMM-C algorithm is applied. Note that there are n binary functions $(\phi_1(\xi_i), \phi_2(\xi_i), \dots, \phi_n(\xi_i))$ needed to be obtained for one specific noise distribution. Thus n auxiliary variables $(\vec{w}_1(\xi_i), \vec{w}_2(\xi_i), \dots, \vec{w}_n(\xi_i))$ are introduced and each component is defined as $\vec{w}_h(\xi_i) = [\vec{w}_{h1}(\xi_i), \vec{w}_{h2}(\xi_i)]^T$ with property $\vec{w}_h \approx \nabla \phi_h(\xi_i)$. Likewise, n Lagrangian multipliers $(\vec{\lambda}_1(\xi_i), \vec{\lambda}_2(\xi_i), \dots, \vec{\lambda}_n(\xi_i))$ are also brought in with $\vec{\lambda}_h(\xi_i) = [\vec{\lambda}_{h1}(\xi_i), \vec{\lambda}_{h2}(\xi_i)]^T$. Based on the convex relaxation method,

(4.22) is rewritten into the following augmented Lagrangian functional

$$\begin{aligned}
& (\theta_{h(\xi_i)}^{k+1}, \phi_{h(\xi_i)}^{k+1}, \vec{w}_{h(\xi_i)}^{k+1}) \\
= & \arg \min_{\theta_h, \phi_h \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Depth}}(\theta_{h(\xi_i)}, \phi_{h(\xi_i)}) \right. \\
= & \sum_{h=1}^n \int_{\Omega} g(\kappa_h(\xi_i)) |\nabla \phi_h(\xi_i)| dx + \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h(\xi_i)) \chi_h(\xi_i) dx \\
& + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}(\xi_i)) \chi_{n+1}(\xi_i) dx + \sum_{h=1}^n \int_{\Omega} v_h^k(\xi_i) \cdot \phi_h(\xi_i) + \frac{\tau}{2} (\phi_h(\xi_i) - \phi_{h(\xi)}^k)^2 dx \\
& \left. + \sum_{h=1}^n \int_{\Omega} \vec{\lambda}_h(\xi_i) \cdot (\vec{w}_h(\xi_i) - \nabla \phi_h(\xi_i)) + \frac{\mu}{2} (\vec{w}_h(\xi_i) - \nabla \phi_h(\xi_i))^2 dx \right\} \tag{4.25}
\end{aligned}$$

where μ is a positive penalty parameter. Here $h = 1, 2, \dots, n$ refer to the number of binary level set functions and $i = 1, 2, \dots, N$ refer to the number of noise distributions. In order to solve (4.25) efficiently with ADMM-C, we first initialize the unknown $\phi_h^0(\xi_i)$, $\vec{w}_h^0(\xi_i)$ and $\vec{\lambda}_h^0(\xi_i)$ at the initial iterative step $k = 0$. Next, some minimization problems are solved with respect to only one kind of unknowns while others are temporarily fixed at each step from k -th to $(k+1)$ -th until convergence is reached. With this alternating direction optimization strategy, the optimization problem (4.25) can be divided into three minimization problems in the iterative process from k -th to $(k+1)$ -th step:

$$\theta_{h(\xi_i)}^{k+1} = \arg \min_{\theta_h} \left\{ \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h(\xi_i)) \chi_h^k(\xi_i) dx + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}(\xi_i)) \chi_{n+1}^k(\xi_i) dx \right\}, \tag{4.26}$$

$$\begin{aligned}
\phi_{h(\xi_i)}^{k+1} = & \arg \min_{\phi \in [0,1]} \left\{ \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h^{k+1}(\xi_i)) \chi_h(\xi_i) dx + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}^{k+1}(\xi_i)) \chi_{n+1}(\xi_i) dx \right. \\
& + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi_i) \phi_h(\xi_i) + \frac{\tau}{2} (\phi_h(\xi_i) - \phi_{h(\xi)}^k)^2) dx \\
& \left. + \sum_{h=1}^n \int_{\Omega} (\vec{\lambda}_h^k(\xi_i) \cdot (\vec{w}_h^k(\xi_i) - \nabla \phi_h(\xi_i)) + \frac{\mu}{2} (\vec{w}_h^k(\xi_i) - \nabla \phi_h(\xi_i))^2) dx \right\} \tag{4.27}
\end{aligned}$$

$$\begin{aligned} \vec{w}_h^{k+1} &= \arg \min_{\vec{w}_h} \left\{ \sum_{h=1}^n \int_{\Omega} g(\kappa_h^{k+1}(\xi_i)) |\vec{w}_h(\xi_i)| dx + \sum_{h=1}^n \int_{\Omega} \vec{\lambda}_h^k(\xi_i) \cdot (\vec{w}_h(\xi_i) - \nabla \phi_h^{k+1}(\xi_i)) \right. \\ &\quad \left. + \frac{\mu}{2} (\vec{w}_h(\xi_i) - \nabla \phi_h^{k+1}(\xi_i))^2 dx \right\}, \quad \text{where } g(\kappa_h^{k+1}(\xi_i)) = \nabla \cdot \frac{\nabla \phi_h^{k+1}(\xi_i)}{|\nabla \phi_h^{k+1}(\xi_i)|_\epsilon}, \end{aligned} \quad (4.28)$$

$$\vec{\lambda}_{h(\xi_i)}^{k+1} = \vec{\lambda}_{h(\xi_i)}^k + \mu (\vec{w}_h^{k+1}(\xi_i) - \nabla \phi_h^{k+1}(\xi_i)). \quad (4.29)$$

To obtain $\theta_h^{k+1} = (\mu_h, \sigma_h)$: The average image intensity values $\mu_h^{k+1}(\xi_i)$ as well as variances $\sigma_h^{k+1}(\xi_i)$ in the foreground and background can be obtained by using the standard variational method based on (4.26), which are given by the following equations (Table 4.1)

Table 4.1: Potential functions of different noise distributions for segmentation with depth application.

Functions	Gaussian noise	Rayleigh noise
$Q_{h'}$ ($h' = 1, \dots, n+1$)	$\frac{1}{2} \log 2\pi + \log \sigma_{h'} + \frac{(f - \mu_{h'})^2}{2\sigma_{h'}^2}$	$2 \log \sigma_{h'} - \log f + \frac{f^2}{2\sigma_{h'}^2}$
Parameters $\theta_{h'} = (\mu_{h'}, \sigma_{h'})$	$\mu_{h'} = \frac{\int_{\Omega} f \chi_{h'} dx}{\int_{\Omega} \chi_{h'} dx}$ $\sigma_{h'}^2 = \frac{\int_{\Omega} (f - \mu_{h'})^2 \chi_{h'} dx}{\int_{\Omega} \chi_{h'} dx}$	$\sigma_{h'}^2 = \frac{\int_{\Omega} f^2 \chi_{h'} dx}{2 \int_{\Omega} \chi_{h'} dx}$
Functions	Poisson noise	Gamma noise
$Q_{h'}$ ($h' = 1, \dots, n+1$)	$\sigma_{h'} - f \log \sigma_{h'}$	$\frac{f}{\mu_{h'}} + \log \mu_{h'}$
Parameters $\theta_{h'} = (\mu_{h'}, \sigma_{h'})$	$\sigma_{h'} = \frac{\int_{\Omega} f \chi_{h'} dx}{\int_{\Omega} \chi_{h'} dx}$	$\mu_{h'} = \frac{\int_{\Omega} f \chi_{h'} dx}{\int_{\Omega} \chi_{h'} dx}$

To obtain ϕ_h^{k+1} : For the minimization problem (4.27) with respect to the function $\phi_h(\xi_i)$, the corresponding Euler-Lagrange equation is given as

$$\begin{aligned} &(-\mu \Delta + \tau) \phi_h(\xi_i) \\ &= -Q_h^*(x, \theta_h^{k+1}(\xi_i)) \prod_{j=1}^{h-1} (1 - \phi_j(\xi_i)) + \tau \phi_h^k(\xi_i) - v_h^k(\xi_i) - \nabla \cdot \vec{\lambda}_h^k(\xi_i) - \mu \nabla \cdot \vec{w}_h^k(\xi_i) \\ &+ \sum_{s=h+1}^{n+1} \{Q_s^*(x, \theta_s^{k+1}(\xi_i)) \phi_s(\xi_i) \prod_{j=1}^{h-1} (1 - \phi_j(\xi_i)) \prod_{j=h+1}^{s-1} (1 - \phi_j(\xi_i))\} \end{aligned} \quad (4.30)$$

Though the above equation is very complicated, FFT can be applied here to accelerate the

calculation. Equation (4.30) can be rewritten as

$$F^*LF\phi_h(\xi_i) = \tau\phi_{h(\xi)}^k - \Lambda_h(\xi_i) - v_h^k(\xi_i) - \nabla \cdot \vec{\lambda}_h^k(\xi_i) - \mu\nabla \cdot \vec{w}_h^k(\xi_i), \quad (4.31)$$

where $L = -\mu F\Delta F^* + \tau$ and F^* is the discrete inverse Fourier transform and

$$\begin{aligned} \Lambda_h(\xi_i) &= Q_h^*(x, \theta_h^{k+1}(\xi_i)) \prod_{j=1}^{h-1} (1 - \phi_j(\xi_i)) \\ &- \sum_{s=h+1}^{n+1} \{Q_s^*(x, \theta_s^{k+1}(\xi_i))\phi_s(\xi_i) \prod_{j=1}^{h-1} (1 - \phi_j(\xi_i)) \prod_{j=h+1}^{s-1} (1 - \phi_j(\xi_i))\}. \end{aligned}$$

Then we can obtain optimal value of $\phi_{h(\xi_i)}$ as follows

$$\phi_{h(\xi_i)}^{k+1} = F^*(L^{-1}F(\tau\phi_{h(\xi)}^k - \Lambda_h(\xi_i) - v_h^k(\xi_i) - \nabla \cdot \vec{\lambda}_h^k(\xi_i) - \mu\nabla \cdot \vec{w}_h^k(\xi_i))). \quad (4.32)$$

To obtain \vec{w}_h^{k+1} : The calculation result of \vec{w} minimization problem (4.28) can be obtained via the generalized soft thresholding formula as

$$\vec{w}_{h(\xi_i)}^{k+1} = \max \left(\left| \nabla\phi_{h(\xi_i)}^{k+1} - \frac{\vec{\lambda}_h^k(\xi_i)}{\mu} \right| - \frac{g(\kappa_h^{k+1}(\xi_i))}{\mu}, 0 \right) \frac{\nabla\phi_{h(\xi_i)}^{k+1} - \frac{\vec{\lambda}_h^k(\xi_i)}{\mu}}{\left| \nabla\phi_{h(\xi_i)}^{k+1} - \frac{\vec{\lambda}_h^k(\xi_i)}{\mu} \right|}. \quad (4.33)$$

The overall algorithm for the Euler's elastica based segmentation with depth in stochastic programming in a pseudo code format is presented as follows.

Algorithm 4.3 Computing framework for (4.19) and (4.20) via PHA

Input: $\phi_h^0(\xi_i)$ ($h = 1, \dots, n$), $p(\xi_i)$, $v_h^0(\xi_i)$, α, β, τ

for $k \geq 1$, do the following steps in turn

- 1: Obtain $\phi_{h(\xi_i)}^{k+1}$ via **Algorithm 4.4**
- 2: Update $\phi_{h(\xi)}^{k+1}$ via Equation (4.23)
- 3: Update $\phi_h^{k+1}(\xi_i)$, $v_h^{k+1}(\xi_i)$ via Equation (4.24)
- 4: **if** some stopping criteria (given in Section 4.3.5) are satisfied **break**

Return optimal value $(\phi_{1(\xi)}^{k+1}, \phi_{2(\xi)}^{k+1}, \dots, \phi_{n(\xi)}^{k+1})$ after thresholding

Algorithm 4.4 Details for step 2 in Algorithm 4.3 via ADMM-C

If $k = 1$

input $\vec{w}_h^0(\xi_i)$, and $\vec{\lambda}_h^0(\xi_i)(h = 1, \dots, n), \mu$

else solve the following problems alternatively

- 1: Update $\theta_{h(\xi_i)}^{k+1}$ according to distribution laws
 - 2: Update $\phi_{h(\xi_i)}^{k+1}$ via minimization problem (4.27)
 - 3: Update $\vec{w}_{h(\xi_i)}^{k+1}$ via minimization problem (4.28)
 - 4: Update $\vec{\lambda}_{h(\xi_i)}^{k+1}$ via (4.29) using gradient ascent method
-

4.2.5 Termination Criteria

The stopping criteria for the entire algorithm are described in this section. As described in [12,165], the iterations will be terminated when the following criteria are satisfied:

- **For the major framework PHA in Algorithm 4.1 and Algorithm 4.3:** During iteration, the constraint errors of $(\phi_{\xi_i} - \phi_{\xi})$, the relative errors of Lagrange multipliers and the optimal solutions should be monitored. They should decrease to a sufficiently small level

$$\text{Algorithm 1} \left\{ \begin{array}{l} R_{\tau}^k = \frac{\sum_{i=1}^N p(\xi_i) \|\phi_{\xi_i}^k - \phi_{\xi}^k\|_{L^1}}{\sum_{i=1}^N p(\xi_i) \|\phi_{\xi_i}^0 - \phi_{\xi}^0\|_{L^1}} \\ R_{v_{\xi}}^k = \frac{\|v_{\xi}^k - v_{\xi}^{k-1}\|_{L^1}}{\|v_{\xi}^{k-1}\|_{L^1}} \text{ with } v_{\xi}^k = \sum_{i=1}^N p(\xi_i) v_{\xi}^k(\xi_i) \\ R_{\phi_{\xi}}^k = \frac{\|\phi_{\xi}^k - \phi_{\xi}^{k-1}\|_{L^1}}{\|\phi_{\xi}^{k-1}\|_{L^1}} \end{array} \right.$$

$$\text{Algorithm 3} \left\{ \begin{array}{l} R_{\tau}^k = \frac{\sum_{h=1}^n \sum_{i=1}^N p(\xi_i) \|\phi_{h(\xi_i)}^k - \phi_{h(\xi)}^k\|_{L^1}}{\sum_{h=1}^n \sum_{i=1}^N p(\xi_i) \|\phi_{h(\xi_i)}^0 - \phi_{h(\xi)}^0\|_{L^1}} \\ R_{v_{h(\xi)}}^k = \frac{\|v_{h(\xi)}^k - v_{h(\xi)}^{k-1}\|_{L^1}}{\|v_{h(\xi)}^{k-1}\|_{L^1}} \text{ with } v_{h(\xi)}^k = \sum_{i=1}^N p(\xi_i) v_h^k(\xi_i) \\ R_{\phi_{h(\xi)}}^k = \frac{\|\phi_{h(\xi)}^k - \phi_{h(\xi)}^{k-1}\|_{L^1}}{\|\phi_{h(\xi)}^{k-1}\|_{L^1}} \end{array} \right.$$

where, $\|\cdot\|_{L^1}$ denotes the L^1 norm on image domain Ω . All components are calculated

in pixel wise. If $R^k < l$ (l is a small enough parameter), the iteration process will be stopped. Note that Equation $R_{v_\xi}^k$ and $R_{v_{h(\xi)}}^k$ can be quite small if the penalty parameters are large. This is due to their explicit dependence on the penalty parameters.

The relative energy error should also be considered. The following form can be used:

$$R_e^k = \|E^k - E^{k-1}\| / \|E^{k-1}\|$$

where $E^k = \sum_{i=1:N} p(\xi_i) E^k(\xi_i)$. The computation terminates when R_e^k is less than a predefined tolerance, which indicates that the energy approaches its steady state.

- **For sub minimization problems using ADMM-C in Algorithm 4.2 and Algorithm 4.4:** The following constraint errors of $(\vec{w}_{\xi_i} - \nabla \phi_{\xi_i}^k)$ and the relative errors of its corresponding Lagrange multipliers in iterations need to be monitored

$$\text{Algorithm 2} \left\{ \begin{array}{l} R_{\vec{w}_\xi}^k = \frac{\sum_{i=1}^N p(\xi_i) \|\vec{w}_{\xi_i}^k - \nabla \phi_{\xi_i}^k\|_{L^1}}{\sum_{i=1}^N p(\xi_i) \|\vec{w}_{\xi_i}^0 - \nabla \phi_{\xi_i}^0\|_{L^1}} \\ R_{\vec{\lambda}_\xi}^k = \frac{\|\vec{\lambda}_\xi^k - \vec{\lambda}_\xi^{k-1}\|_{L^1}}{\|\vec{\lambda}_\xi^{k-1}\|_{L^1}} \text{ with } \vec{\lambda}_\xi^k = \sum_{i=1}^N p(\xi_i) \vec{\lambda}_\xi^k(\xi_i) \end{array} \right.$$

$$\text{Algorithm 4} \left\{ \begin{array}{l} R_{\vec{w}_{h(\xi)}}^k = \frac{\sum_{h=1}^n \sum_{i=1}^N p(\xi_i) \|\vec{w}_{h(\xi_i)}^k - \nabla \phi_{h(\xi_i)}^k\|_{L^1}}{\sum_{h=1}^n \sum_{i=1}^N p(\xi_i) \|\vec{w}_{h(\xi_i)}^0 - \nabla \phi_{h(\xi_i)}^0\|_{L^1}} \\ R_{\vec{\lambda}_{h(\xi)}}^k = \frac{\|\vec{\lambda}_{h(\xi)}^k - \vec{\lambda}_{h(\xi)}^{k-1}\|_{L^1}}{\|\vec{\lambda}_{h(\xi)}^{k-1}\|_{L^1}} \text{ with } \vec{\lambda}_{h(\xi)}^k = \sum_{i=1}^N p(\xi_i) \vec{\lambda}_h^k(\xi_i) \end{array} \right.$$

All numerical quantities are presented in log scale. Some specific methods are used to tune parameters in the implementation process of the proposed approach. The two parameters in $g(\kappa) = \alpha + \beta|\kappa|$, α and β , control the length and curvature of the segmentation boundary. The ratio between a and b is related to the connectivity and smoothness of the level lines. As discussed in [212], the connection of disconnected level lines and smoothness of level lines can be guaranteed by a large parameter β . In addition, how to determine the other two parameters: τ and μ associated with Lagrange multipliers will be illustrated. Numerical indicators give the basis of penalty parameter adjustment. One example of their values selection is given in Experiment 4.4.1.

Lastly, the convergency of PHA needs to be considered. PHA is applied to the general stochastic programming problems of two-phase image segmentation and segmentation with depth in gray and color images, as formulated in (4.4) and (4.21). It is implemented relying on the convex minimization for each scenario ξ_i (given in Algorithm 4.1 and Algorithm 4.3). In this way, the sequence of pairs (ϕ_ξ^k, v_ξ^k) generated from PHA are sure to converge to a solution pair. Please refer to [134] for discussion on the convergence of PHA theoretically with clear proof.

4.3 Experiments

The proposed segmentation formulations and developed algorithm extensively on various synthetic and real images for multiple purposes are applied. Experimental results are used to validate the performance and efficiency of our proposed models and algorithm. All the experiments are implemented on the same platform (Matlab 8.2) on a PC (Intel (R), CPU: 2.80GHz, RAM: 16GB, cores number: 4, architecture: 64-bit).

4.3.1 Experiments for Two-Phase Segmentation on Synthetic Images

Some synthetic images of size 256×256 pixels are used as the test images. In these experiments, two-phase CV model [40] and the CVE model [212] are used for comparison in order to show the performance of our proposed model. The reason of choosing [40] and [212] for comparison is that they are both classic models in variational segmentation with excellent results. Firstly, we set $v^0(\xi_i) = 0$, $\vec{w}^0(\xi_i) = \vec{0}$, $\vec{\lambda}^0(\xi_i) = \vec{0}$ and all the Lagrange multipliers are initially set to be 0 for all the numerical experiments. The same initialization of variables in each experiment are used in order to have a relatively fair comparison. In Fig. 4-2, results of the CV model, CVE model and our proposed model applied on noisy images are presented. The original images, noisy images with stochastic noises including the Gaussian noise, Rayleigh noise, Poisson noise and Gamma noise, and initialization for ϕ^0 are shown in (a) and (e). In addition, the pepper & salt noise is additionally contained in (a). Results obtained by the CV model are presented in (b) and (f), by CVE model in (c) and (g), and from our model (4.2) in (d) and (h) respectively. From left to right in (b)-(d) and (f)-(h), we start with the optimal solution ϕ_ξ , followed by final curves plotted on noisy images (red lines) and final curves plotted individually (blue lines). It helps to distinguish the detailed

differences between results obtained from different models by presenting the final results in blue separately.

It can be clearly seen that the results of (d) and (h) obtained by our model (4.2) are much better than those from the other two models. It is known that CV model is robust against general noises. However when the noise level is getting much higher, CV model fails to obtain the desired results as shown in (b) and (f). It completely fails since it is driven by the mean level of the target region resulting in the fact that it cannot recognize whether one particular pixel belongs to big noises or objects. CVE model has better performance as shown in (c) for segmentation in noisy images as well as capturing meaningful structures. However it ignores some relatively small objects, thus some small and homogeneous parts of objects can not be captured as shown in (g). The CVE model fails there when the desired object has similar density as the background. With the increase of the homogeneity, this kind of drawback becomes more obvious.

It should be noted that all the noises contained in the images are random in our experiments, which means we do not need to set specific percentage for each type of noise. The noise level of one specific noise type could be 0%. The parameters for CV model, CVE model and our proposed model are given as follows

CV model [40]		CVE model [212]	
Fig. 4-2(b):	$\mu = 20, \gamma = 3$ $\alpha_1 = 10, \alpha_2 = 10$	Fig. 4-2(c):	$\alpha = 3, \beta = 15, \mu = 20$ $\alpha_1 = 8, \alpha_2 = 8$
Fig. 4-2(f):	$\mu = 20, \gamma = 3$ $\alpha_1 = 7, \alpha_2 = 7$	Fig. 4-2(g):	$\alpha = 3, \beta = 10, \mu = 20$ $\alpha_1 = 8, \alpha_2 = 8$
Our proposed model (4.2) via PHA with ADMM-C			
Fig. 4-2(d):	$\alpha = 3, \beta = 25, \tau = 5, \mu = 20$ $\alpha_1 = 10, \alpha_2 = 10, p(\xi) = (0.4, 0.1, 0.3, 0.2)$		
Fig. 4-2(h):	$\alpha = 3, \beta = 25, \tau = 5, \mu = 20$ $\alpha_1 = 7, \alpha_2 = 7, p(\xi) = (0.4, 0.1, 0.3, 0.2)$		

Fig. 4-3 demonstrates the performance of model 4.2 without the last two terms ($v(\cdot)$ term and τ -term). Some segmentation results obtained from our model (4.2) under this assumption are given in Fig. 4-3(a) and (b). It can be observed that the segmentation results are very close to the ones obtained when the last two terms are presented. However, the convergence cannot be guaranteed without these two terms in general. Therefore, the similar performances with or without the last two terms are only for special images, which cannot be guaranteed in general without convergence analysis. Technically, $w(\cdot)$ serves as

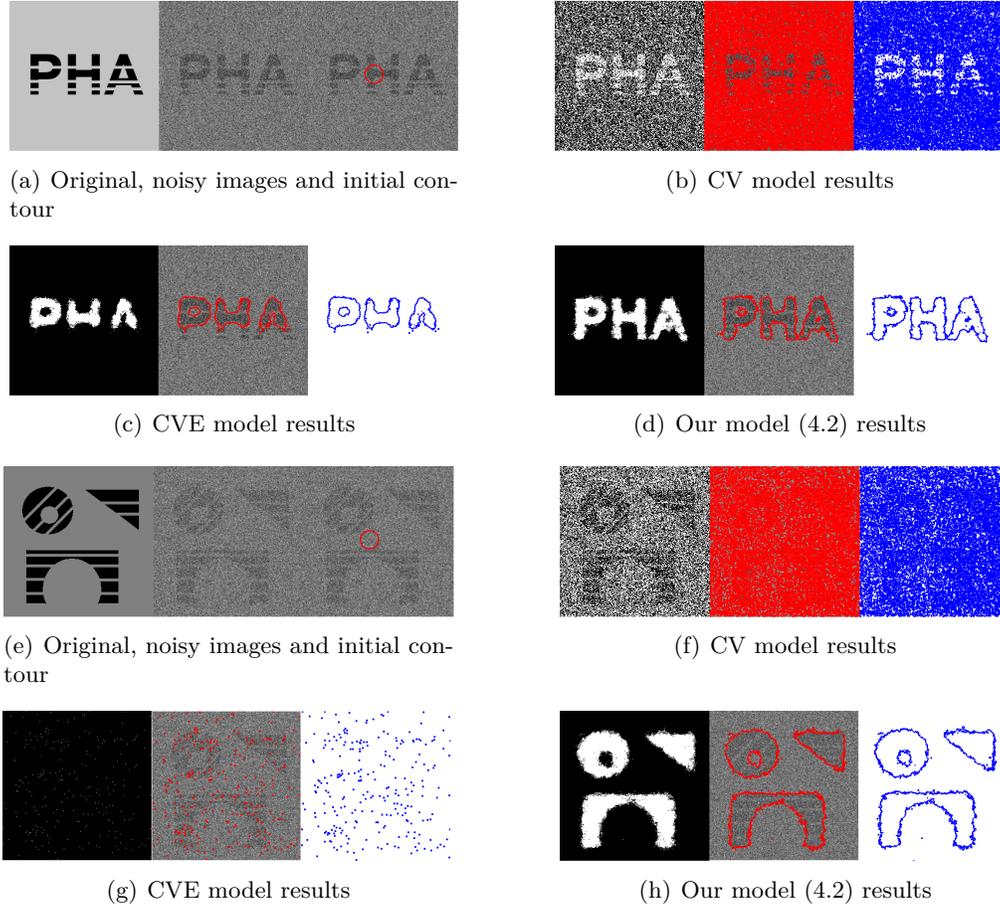
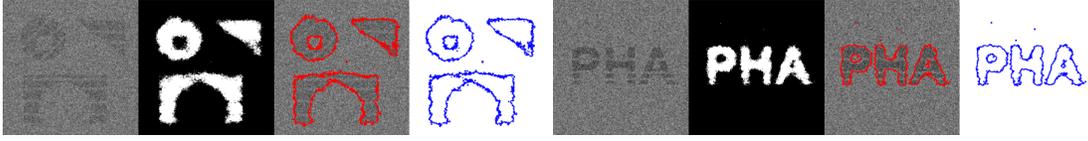


Figure 4-2: Two phase segmentation for synthetic noisy images with incomplete shapes and letters. (a) and (e): original images, noisy images and the same initial ϕ^0 ; (b) and (f): results obtained by CV model; (c) and (g): results obtained by CVE model; (d) and (h): final results from our proposed model (4.2).

the multiplier vector for the non-anticipativity constraint. Each of the w -iterates is forced to stay in space \mathcal{M} , thus the existence and uniqueness of the solution will be guaranteed. The r -term makes the related minimization sub-problems be strongly convex. These two terms are of great significance for our theoretical analysis as detailed in Section 4.2.3.

Fig. 4-4 gives an example that illustrates the convergence of the relative residuals given in Section 4.3.5 (R_τ^k and $R_{\bar{w}_\xi}^k$ in *Algorithm 4.1* & *4.3*), the relative errors of Lagrange multipliers ($R_{v_\xi}^k$ and $R_{\lambda_\xi}^k$ in *Algorithm 4.1* & *4.3*), the relative error of $R_{\phi_\xi}^k$ (in *Algorithm 4.1*) and the relative energy error R_e^k in our model respectively. They are obtained for the image in Fig. 4-2(h). It is clearly shown that the proposed algorithm has converged well before 100 iterations. They also give an important clue on how to choose the penalty



(a) Results from (4.2) without $v(\cdot)$ term and r -term (b) Results from (4.2) without $v(\cdot)$ term and r -term

Figure 4-3: Segmentation results from our model (4.2) without the last two terms ($v(\cdot)$ term and r -term).

parameters t and m . In order to ensure convergence as well as achieving a high speed of convergence, the errors R_τ^k , $R_{\bar{w}}^k$, R_v^k and R_λ^k should converge steadily with nearly the same speed. If R_τ^k , $R_{\bar{w}}^k$ go to zero faster than the others, τ and μ can be decreased and vice versa. R_τ^k , $R_{\bar{w}}^k$ will converge to zero with the same speed as the iteration proceeds and the energy will decrease to a steady constant value when τ and μ are chosen properly.

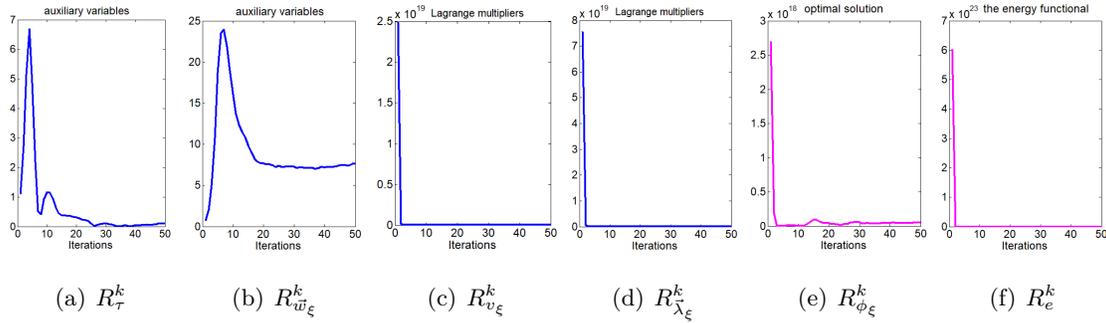


Figure 4-4: The relative residual plots according to the equations in Section 4.3.5. (a): auxiliary variables in R_τ^k ; (b): auxiliary variables in $R_{\bar{w}}^k$; (c): Lagrange multipliers in R_v^k ; (d): Lagrange multipliers in R_λ^k ; (e): optimal solution $R_{\phi_\xi}^k$; (f): the energy functional.

4.3.2 Experiments for Two-Phase Segmentation on Real Images

A tiger image of size 481×321 and a butterfly image of size 230×137 are used as the test images. Visual comparisons with the results and the evolution process of our model are provided. In Fig. 4-5, results of the color CV model [37], color CVE model and our proposed model (4.3) are shown respectively. The color CVE model is stated as

$$\begin{aligned}
 E(\phi, c) &= \alpha_1 \int_{\Omega} \sum_{l=1}^m (f_l - c_{1l})^2 \phi dx + \alpha_2 \int_{\Omega} \sum_{l=1}^m (f_l - c_{2l})^2 (1 - \phi) dx \\
 &+ \int_{\Omega} (\alpha + \beta \kappa^2) |\nabla \phi| dx.
 \end{aligned} \tag{4.34}$$

Here we extend CVE model from gray scale to color scale according to the coupling approach used in [37]. Fig. 4-5(a), (g) show the real noisy images. (b), (h) give the initialization of ϕ^0 . The segmented images by using the color CV model are shown in (c) and (i). Fig. 4-5(d) and (j) show the results from the color CVE model. Results from our proposed model are shown in (e) and (k). (f) and (l) show the intermediate evolution process of the contour ϕ_ξ obtained from our proposed model. The results from color CV model are chosen as the input of ϕ^0 . ϕ^0 can also be initialized randomly while the final results vary little. One feature in the images is the discontinuous stripes on the tiger’s tail or the sparse spots on the butterfly’s wings. With the proceeding of iteration, it can be observed that CV model fails to capture the correct boundaries of the objects, while both CVE and our model are able to complete an intact shape regardless of the existing gaps within the objects. However, CVE model is inevitably influenced by the stochastic noises especially when these noises smear the elongated structures like the tiger’s tail or the level of homogeneity increases. The parameters used to obtain Fig. 4-5(c)-(e) and (i)-(k) are

Color CV model [37]		Color CVE model (4.34)	
Fig. 4-5(c):	$\mu = 3, \gamma = 3$ $\alpha_1 = 8, \alpha_2 = 5$	Fig. 4-5(d):	$\alpha = 3, \beta = 8, \mu = 20$ $\alpha_1 = 5, \alpha_2 = 5$
Fig. 4-5(i):	$\mu = 3, \gamma = 3$ $\alpha_1 = 8, \alpha_2 = 5$	Fig. 4-5(j):	$\alpha = 3, \beta = 20, \mu = 20$ $\alpha_1 = 7, \alpha_2 = 7$
Our proposed model (4.3) via PHA with ADMM-C			
Fig. 4-5(e):	$\alpha = 3, \beta = 4, \tau = 5, \mu = 80$ $\alpha_1 = 4, \alpha_2 = 4, p(\xi) = (0.6, 0.1, 0.2, 0.1)$		
Fig. 4-5(k):	$\alpha = 3, \beta = 16, \tau = 3, \mu = 50$ $\alpha_1 = 4, \alpha_2 = 4, p(\xi) = (0.6, 0.1, 0.2, 0.1)$		

4.3.3 Experiments for Segmentation with Depth on Synthetic Images

The proposed models (4.19) and (4.20) using PHA will be compared with classic segmentation with depth model (2.16) on synthetic images without stochastic programming. The detailed implementation framework is shown as follows and all experiments for segmentation with depth follow the same procedure.

In order to speed up the evolution of contours and improve the accuracy of results, we initialize the binary level-set functions $\phi_h^0(\xi_1) = \dots = \phi_h^0(\xi_N)$ using the results from the the

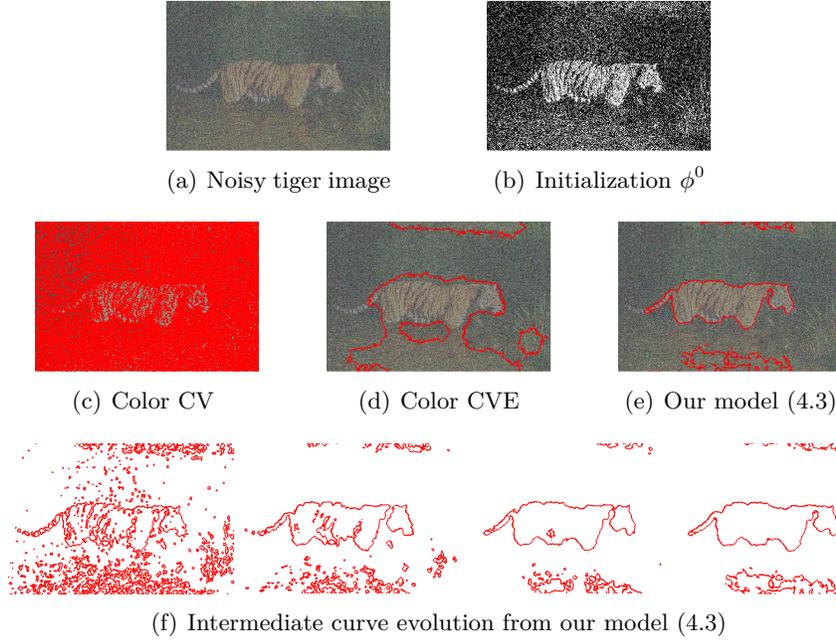


Figure 4-5: Two phase segmentation for real noisy images with incomplete shapes. (a) and (g): noisy images; (b) and (h): initial ϕ^0 ; (c) and (i): results obtained by CV model; (d) and (j): results obtained by CVE model; (e) and (k): final results from our proposed model (4.3); (f) and (l): intermediate curve evolution by our proposed model.

standard multiphase segmentation model in stochastic programming, which is given by

$$\begin{aligned}
& \arg \min_{\theta_\xi, \phi_\xi \in \{0,1\}} \left\{ E_{\text{SP-general}}^{\text{Multi}}(\theta_\xi, \phi_\xi) \right. & (4.35) \\
& = \sum_{h=1}^n \int_{\Omega} |\nabla \phi_h(\xi)| dx + \sum_{h=1}^n \int_{\Omega} Q_h^*(x, \theta_h(\xi)) \chi_h(\xi) dx + \int_{\Omega} Q_{n+1}^*(x, \theta_{n+1}(\xi)) \chi_{n+1}(\xi) dx \\
& \left. + \sum_{h=1}^n \int_{\Omega} (v_h^k(\xi) \cdot \phi_h(\xi) + \frac{\tau}{2} (\phi_h(\xi) - \phi_{h(\xi)}^k)^2) dx \right\}
\end{aligned}$$

Different from the traditional ones [166, 210] where contours are initialized by the standard multiphase segmentation model without stochastic programming, we take the situation of unknown noises into consideration. In experiments, it can be observed that the initialization of $\phi_h^0(\xi_i)$ will be inevitably affected when big stochastic noises are contained in the original image, which may lead to a failure for entire framework to obtain expected results. An example is shown below in Fig. 4-6. Fig. 4-6(a) shows the synthetic image (size 100×100) with two circles corrupted by random noises and the results from standard multiphase segmentation. (b) shows the initialization of the binary level set functions us-

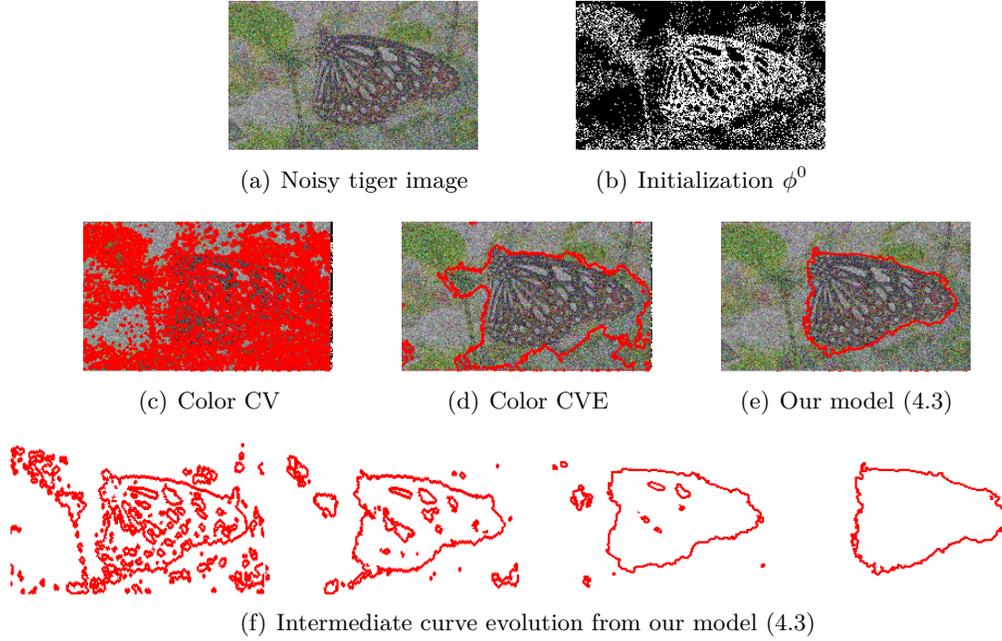
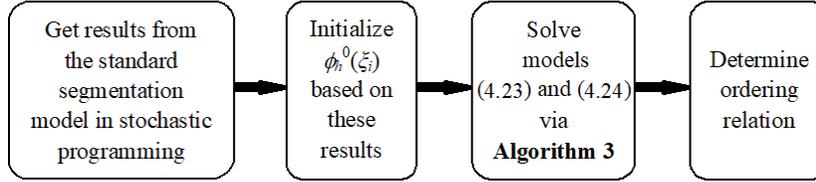


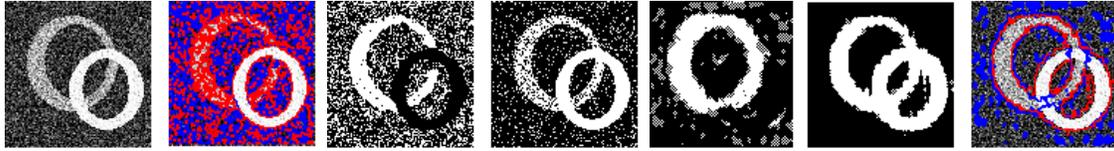
Figure 4-5 continued.



ing results from the standard segmentation method. The results obtained by traditional segmentation with depth model [166, 210] are presented in (c). It can be observed that the traditional implementation framework do not work under the impact of big noises. Furthermore, the CVE model was designed only for one foreground shape recovery without considering segmentation with depth information, which is hence not applicable here.

Fig. 4-7 presents our novel framework for segmentation with depth in stochastic programming. The same testing image is used. With the application of (4.35), we obtain separate objects shown in (a) and initialize ϕ_h^0 in (b). Obvious advantage of our proposed model (4.19) compared with the traditional framework is shown in (c), where the final shapes are reconstructed successfully even though there existing big noises. The parameters used for our proposed model (4.19) are

Our proposed model (4.19) via PHA with ADMM-C
Fig. 4-7(c): $\alpha = 3, \beta = 10, \tau = 5, \mu = 30, p(\xi) = (0.5, 0.1, 0.2, 0.2)$



(g) Noisy image and standard multiphase segmentation results (h) Initial ϕ_h^0 based on standard multiphase segmentation results (i) traditional segmentation with depth results

Figure 4-6: Traditional segmentation with depth for an image with two circles. (a): noisy images and results obtained by the standard multiphase segmentation model; (b): the initialization for two binary functions ϕ_h^0 ; (c): results obtained by traditional segmentation with depth model [13,15].



(a) Noisy image and results obtained from traditional (4.35) (b) Initial ϕ_h^0 for two binary functions (c) Our proposed model (4.19) results

Figure 4-7: Our proposed model (4.19) for an image with two circles. (a): noisy images and results by the standard multiphase segmentation in stochastic programming model; (b): the initialization for two binary functions ϕ_h^0 ; (c): results obtained by our proposed model

In order to determine the ordering relations of the white circle and gray circle, we minimize the energy functional (4.19) based on the assumptions that the white circle is occluded by the gray circle or the gray circle is occluded by the white circle. The results are listed in Table 4.2, from which we can deduce that the white circle, the gray circle and the background are ordered from the nearest to farthest with respect to the observer.

Table 4.2: Minimal energies of different ordering relations.

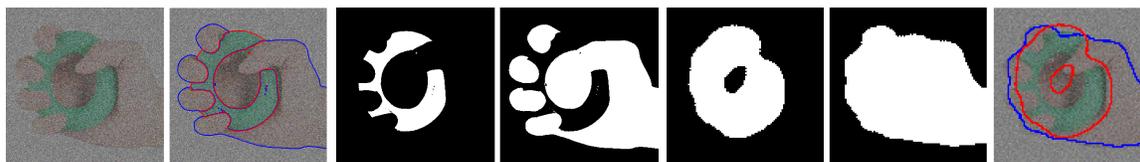
Possible Order	Minimum of energy functional
1. white circle \Rightarrow gray circle	18.56
2. gray circle \Rightarrow white circle	20.83

4.3.4 Experiments for Segmentation with Depth on Real Images

In the last experiment, a real image with a circle and a hand (size 360×360) and the other with a bird and a trunk (size 220×241) are shown in Fig. 4-8. Fig. 4-8(a) and (d) show the original noisy image and the result from the standard segmentation model in stochastic

programming plotted on the original noisy image. The two initial values for ϕ_h^0 are given in (b) and (e). The final results from our model (4.20) are provided in (c) and (f). It can be clearly seen that our model can perform well in real images. The parameters for our proposed model (4.20) are given as follows

Our proposed model (4.20) via PHA with ADMM-C	
Fig. 4-8(c):	$\alpha = 3, \beta = 25, \tau = 5, \mu = 20, p(\xi) = (0.4, 0.1, 0.3, 0.2)$
Fig. 4-8(f):	$\alpha = 3, \beta = 25, \tau = 3, \mu = 10, p(\xi) = (0.5, 0.1, 0.3, 0.1)$



(a) Noisy image and results obtained from functional (4.35) (b) Initial ϕ_h^0 based on results from functional of functional (4.35) (c) Our proposed model (4.20) results



(d) Noisy image and results obtained from functional (4.35) (e) Initial ϕ_h^0 based on results from functional of functional (4.35) (f) Our proposed model (4.20) results

Figure 4-8: Our proposed model (4.20) for real image. (a) and (d): noisy images and results by the standard multiphase segmentation in stochastic programming model; (b) and (e): the initialization for two binary functions ϕ_h^0 ; (c) and (f): results by our proposed model.

To determine the ordering relations in Fig. 4-8(c) and (f), we minimize the energy functional based on the assumptions including all the potential ordering respectively. From the results listed in Table 4.3 and Table 4.4, we can choose the correct orderings which are mapping to the minimal functional values.

Table 4.3: Minimal energies of different ordering relations for Fig. 4-8(c).

Possible Order	Minimum of energy functional
1. green circle \Rightarrow hand	40.57
2. hand \Rightarrow green circle	45.26

There is one important thing to be noted here. When dealing with the convex optimization problem, we have to use a threshold method to realize the binarization of $\phi_{h(\xi)}^{k+1}$.

Table 4.4: Minimal energies of different ordering relations for Fig. 4-8(f).

Possible Order	Minimum of energy functional
1. bird \Rightarrow trunk	28.56
2. trunk \Rightarrow bird	20.83

It is an important way to help find the accurate results. The histograms of non-threshold and threshold results from Fig. 4-8(c) are given in Fig. 4-9. It provides a good way to choose the threshold of ϕ_h^{k+1} . From experiment, we find that the threshold $\eta = 0.5$ could be applicable.

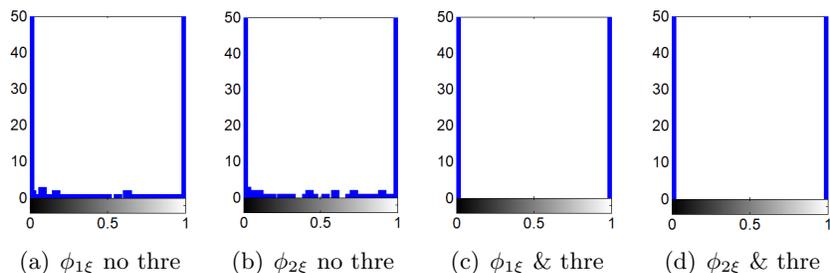


Figure 4-9: Histograms of the binary level set functions (non-threshold and threshold).

At last, the efficiency of our proposed PHA with ADMM-C algorithm is demonstrated by the number of iterations and computation time in Table 4.5. The iterations and time are shown according to Experiments 4.3.1 to 4.3.4. The computational time is measured in seconds.

Table 4.5: Number of iterations and computational time.

Image	Size	Iterations	Time
Fig. 4-2(d)	256×256	73	1.25
Fig. 4-2(h)	256×256	80	1.53
Fig. 4-5(e)	481×321	55	7.6
Fig. 4-5(k)	230×137	50	0.79
Fig. 4-7(c)	100×100	45	1.23
Fig. 4-8(c)	360×360	60	8.9
Fig. 4-8(f)	220×241	55	2.69

4.4 Summary

Euler’s elastica-based segmentation models have strong capability of completing the missing boundaries for existing objects in a clean image, but they are not working well for noisy

images. In this chapter a Euler’s elastica-based approach is established that can properly deal with random noises to improve the segmentation performance for noisy images. The corresponding formulation of stochastic optimization is solved via PHA, and the description of each individual scenario is obtained by the alternating direction method of multipliers. Technically, sub-problems derived from the framework of PHA can be solved using the curvature-weighted approach and the convex relaxation method. Then, an alternating optimization strategy is applied using some powerful accelerating techniques including the fast Fourier transform and generalized soft threshold formulas. Experiments have been conducted on both synthetic and real images, which displayed significant performance gains of the proposed segmentation models and demonstrated the advantages of the developed algorithms. In the next chapter, we will turn into the third focus of this thesis, the methodological investigation on incorporation between variational methods and deep networks for further performance improvements.

Chapter 5

Multi-task Deep Variational Model for Denoising, Inpainting, Decomposition and Segmentation

5.1 Introduction

For image processing approaches, it is desirable to generate outputs without losing important features (i.e. edge sharpness, smoothness, corner preservation and contrast etc.). As illustrated in [160,163,164,166,189], variational image processing works very well in preserving important image features, dealing with situations such as large intensity inhomogeneity etc. and requires less memory in computation. One stunning property of variational methods is that it can be easily combined with other advanced theories or incorporated into existing classic frameworks. For example, underwater dark channel prior method and nonlocal means were incorporated into variational models for underwater image restoration [73,74], and stochastic programming was applied for image segmentation with random noises [162], results with improved performance were obtained and wider applicability was achieved.

In recent years, deep learning network is used with great success in computational imaging. A huge amount of research was conducted on natural images resulting in significant progress and promising outcomes. Among them, convolutional neural network (CNN) based techniques have made significant achievements for a wide range of applications in image processing problems. For example, Jin et al. [80] proposed a deep convolutional network

for image restoration problem with a focus on biomedical imaging with addition of residual learning to enhance the training propagation of the visual information. Authors in [203] first trained a set of effective CNN denoiser priors for denoising and then set them as plug-in modules to tackle various inverse problems such as image deblurring and super-resolution. Most of the state-of-the-art performance by CNNs for image processing problems is achieved based on large training datasets. Thus it is reasonable to deduce that the excellent capability of learning realistic image priors from data plays a significant role in producing such satisfactory outcomes in CNNs. In fact, learning is only one of the main reasons for the good performance of deep networks. A good example was given in [201], which demonstrated that the structure of the network should correspond well with the structure of the data. Similarly, authors in [170] put forward that image statistics can be adequately captured by the structure of a single CNN generator network in a totally unsupervised way. A unified framework [58] was built on top of [170] by coupling multiple CNN generator networks to handle a wide variety of seemingly unrelated computer vision tasks.

Several researches have showed the great potential for integrating these two excellent techniques to form deep variational models. The recurrent neural network (RNN) was employed in [91] to form the iterative curve deformation process for image segmentation. Researches [29, 90] continued the idea of RNN and extended it to recurrent-CNN architectures for medical image segmentation, which achieved state-of-the-art results. They both modeled each time-step as a feedforward neural network architecture. The difference is that the feature map in [90] was converted from a fully convolutional neural network (FCN) model to a variational level set function, while [29] kept the curve evolution and modeled the corresponding terms in the variational energy by two separate CNNs inspired from the multi-scale image pyramid. According to [44], the loss function was creatively developed with classic variational segmentation model containing contour length and region information for the dense deep learning model optimization. These works were all designed for supervised learning with specific purposes. Although authors in [98] proposed to use Total Variation (TV) [140] as the regularization to improve the Deep Image Prior (DIP) [170] framework. However, only one generator was used in DIP to produce the prior of the original image, which is not sufficient when the original image was separated into several image components.

As DIP [170] can adequately capture the image statistics by the structure of a single CNN generator network independent of learning, double-DIP [58] showed the power of the combination of multiple DIPs by regarding an image as a mixture of its basic layers separately. The main interesting feature of variational methods is that the energy functional with multiple variables utilizing different levels of image information is established in a self-supervised manner. All these properties provide a promising potential for the integration of these two approaches. Our proposed deep variational models may not work as well as some methods particularly designed for a specific task. However, we believe that this is the first work that unifies CNN architectures in a variational way for a wide variety of image tasks.

For the incorporation of deep-image-prior networks and variational image processing models, our contributions can be briefly summarized in the following aspects:

- We propose a novel general model for multi-purpose unsupervised image tasks. An integration approach is proposed, redesigning the variational model and DIPs to generate the output of each variable without iteration. Moreover, this general model provides diverse novel selections of inductive bias for unsupervised learning, which is stemmed from variational tools possessing self-supervision.
- The outstanding feature of our work is the simultaneous permission of multitasking, which is different from other existing multi-purpose deep models. Furthermore, we explore the potential on decomposition based image multiscale segmentation. Technically, high-order terms widely used in variational methods are incorporated for compensating the problems contained in the TV term. They are applied in ℓ_1 -norm penalty to promote sparsity. The effects of these terms can further improve the performance.
- With the incorporation of deep neural networks, limitations of the traditional iteration based fast algorithms [162, 164, 166], being sensitive to numerous pre-defined parameters and heavily dependent on initial settings, can be overcome. In contrast to the classical variational approach, the energy functional serves as the loss function, and the minimization problem is directly achieved using a neural network by back-propagation without any optimization algorithm design.

The rest of this chapter is structured as follows. Section 5.2 shortly reviews the related approaches in this field. Our proposed approach and the general framework are presented in Section 5.3. The experiments conducted with performance evaluation and comparison are described in Section 5.4, followed by the conclusion in Section 5.5.

5.2 Related Work

In this section, existing researches providing fundamental models for our work and used for comparison are presented.

5.2.1 Variational Methods

In this subsection, the fundamental models in the field of variational methods including five high-order models (all in denoising form as an example) and the Vese-Osher model for decomposition are described first, then the edge weighted second order decomposition model used for comparison is briefly introduced.

High-order variational models

Several well-known high-order variational models for image processing are reviewed in this section, which are, Total Laplace (TL), Bounded Hessian (BH), Total Generalized Variation (TGV), Euler's Elastica (EE) and Total Curvature (TC).

Total Laplace. TL model [193] initially proposed for image denoising can be stated as the minimization of the following energy functional

$$E(u) = \frac{1}{2} \int_{\Omega} (f - u)^2 dx + \alpha \int_{\Omega} |\Delta u| dx, \quad (5.1)$$

where f is an observed image defined on the image domain Ω , u is the restored image from f . " Δ " is the laplace operator, which is defined as $\Delta u = \nabla \cdot \nabla u = \nabla \cdot \left(\frac{\partial_x u}{\partial_y u} \right) = \partial_x \partial_x u + \partial_y \partial_y u$. " ∇ " is the gradient operator and " $\nabla \cdot$ " is the divergence operator respectively.

Bounded Hessian. Authors [89, 105] investigated the properties derived from Hessian

matrix and formulated a Hessian regularizer into the energy functional

$$\begin{cases} E(u) = \frac{1}{2} \int_{\Omega} (f - u)^2 dx + \alpha \int_{\Omega} |\nabla^2 u| dx \\ \text{with } \nabla^2 u = \begin{pmatrix} \partial_x \partial_x u & \partial_y \partial_x u \\ \partial_x \partial_y u & \partial_y \partial_y u \end{pmatrix} \end{cases} \quad (5.2)$$

Total Generalized Variation. As described in [18, 54], TGV formulation is able to reconstruct piecewise polynomial functions of arbitrary order (piecewise constant, piecewise affine, piecewise quadratic etc.), which is of the following form

$$\begin{cases} E(u, \vec{p}) = \frac{1}{2} \int_{\Omega} (f - u)^2 dx + \alpha \int_{\Omega} |\nabla u - \vec{p}| dx + \beta \int_{\Omega} |\varepsilon(\vec{p})| dx \\ \text{with } \varepsilon(\vec{p}) = \begin{pmatrix} \partial_x p_1 & \frac{\partial_y p_1 + \partial_x p_2}{2} \\ \frac{\partial_y p_1 + \partial_x p_2}{2} & \partial_y p_2 \end{pmatrix} \end{cases} \quad (5.3)$$

where $\varepsilon(\vec{p})$ is the symmetrised derivative in an auxiliary vector field and \vec{p} is an auxiliary variable such that $\vec{p} \approx \nabla u$. The balance between the first and second derivative of the image information is controlled by the ratio of positive weights α and β . It can be observed that TGV expression is similar to Hessian (5.2), an illustration in terms of the relationship between BH (5.2) and TGV (5.3) was given in [95]. Especially when $\vec{p} = \nabla u$, then the term $|\nabla u - \vec{p}|$ of (5.3) equals 0 and $\varepsilon(\vec{p})$ turns into the BH term $\nabla^2 u$ of (5.2) as a result. Hence we can obtain: $|\varepsilon(\vec{p})|_{TGV} \leq |\nabla^2 u|_{BH}$, which means TGV could be a better choice due to a faster minimization process. More relevant information can be found in [95].

Euler's Elastica. The EE model was presented subsequently in [160] with the formulation as follows

$$E(u) = \frac{1}{2} \int_{\Omega} (f - u)^2 dx + \int_{\Omega} (\alpha + \beta |\nabla \cdot (\frac{\nabla u}{|\nabla u|})|) |\nabla u| dx. \quad (5.4)$$

where κ is the curvature of a 2D curve. ℓ_2 -norm of the curvature was used in [160] in the elastica regularizer as: $(\alpha + \beta \kappa^2) |\nabla u|$. The ℓ_1 -norm of the curvature based regularizer $(\alpha + \beta |\kappa|) |\nabla u|$ was suggested in [166] and [206]. From [166], it can preserve the object corners better than the ℓ_2 form when conducting segmentation. In recent research [206] (they set $\alpha = 1$), it was also called the Total Absolute Curvature with better performance in denoising, segmentation and inpainting over TV [140] and Euler's elastica- ℓ_2 [160] model.

Total Curvature. More recently, the ℓ_1 -norm of curvature, i.e. total curvature [165, 211] was used to design a regularization term for the variation image restoration model in gray image denoising, and in this case, authors in [165] use the following form

$$E(u) = \frac{1}{2} \int_{\Omega} (f - u)^2 dx + \alpha \int_{\Omega} \left| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) \right| dx. \quad (5.5)$$

Here the mathematical expression of κ has been given in (5.4). As demonstrated in [165], the first term of the right side (the data fidelity term) based on ℓ_1 -norm can preserve image contrast as well as remove outliers and irregularities. And the second part (the TC regularization term) is to smooth image while preserve edges and corners of objects without staircase artefact.

Several high order variational models are introduced in chronological order in this section: from the earliest and simplest formulation TL to the latest one TC. The propositions of them lead to the advantages of eliminating the staircase artefact with more natural piecewise planar images obtained as results for image processing and tackling the problem of the edge or corner smearing over their first order counterparts. One well-known type of their first order counterparts is the Total Variation (TV) [140]. All high-order models mentioned above and their specific representations of energy functionals are summarized in Table 5.1. Please note that we introduce these models in denoising formulation as example. In Table 5.1, different potential functions $\Psi(u)$ can be given to describe diverse image tasks: for denoising, $\Psi(u) = (f - u)^2$; for inpainting [160, 165], $\Psi(u) = (m \cdot (f - u))^2$, $m = \begin{cases} 0 & x \in D \\ \lambda_D & x \in \Omega/D \end{cases}$ is the mask function to represent the broken region D .

Table 5.1: High-order Energy Functionals

No.	Name	Functional
1.	TL	$E(u) = \frac{1}{2} \int_{\Omega} \Psi(u) dx + \alpha \int_{\Omega} \Delta u dx$
2.	BH	$E(u) = \frac{1}{2} \int_{\Omega} \Psi(u) dx + \alpha \int_{\Omega} \nabla^2 u dx$
3.	TGV	$E(u, \vec{p}) = \frac{1}{2} \int_{\Omega} \Psi(u) dx + \alpha \int_{\Omega} \nabla u - \vec{p} dx + \beta \int_{\Omega} \varepsilon(\vec{p}) dx$
4.	EE	$E(u) = \frac{1}{2} \int_{\Omega} \Psi(u) dx + \int_{\Omega} (\alpha + \beta \nabla \cdot (\frac{\nabla u}{ \nabla u })) \nabla u dx$
5.	TC	$E(u) = \frac{1}{2} \int_{\Omega} \Psi(u) dx + \alpha \int_{\Omega} \left \nabla \cdot \left(\frac{\nabla u}{ \nabla u } \right) \right dx$
No.	Task Name	Form of $\Psi(u)$
(i)	denoising	$\Psi(u) = (u - f)^2$
(ii)	inpainting	$\Psi(u) = (m \cdot (u - f))^2$, $m = \begin{cases} 0 & x \in D \\ \lambda_D & x \in \Omega/D \end{cases}$

The VeseOsher (VO) model

The main purpose of variational decomposition is to split an image f into the structural component u and textural component v : $f = u + v$, where u contains the main information of geometric features like object edges and hues and v represents the oscillating patterns including both textures and noise. It is a critical operation that makes it easier for subsequent image manipulation such as detail enhancement, visual abstraction, scene understanding, etc. Vese and Osher [172] proposed the VO model by using two functions $\vec{g} = [g_1, g_2]^T$ to define the textures, and the energy functional reads

$$E(u, \vec{g}) = \frac{1}{2} \int_{\Omega} (f - u - \nabla \cdot \vec{g})^2 dx + \alpha \int_{\Omega} |\nabla u| dx + \beta \left[\int_{\Omega} |\vec{g}|^p dx \right]^{\frac{1}{p}}. \quad (5.6)$$

This is a variant of the decomposition functional proposed by Meyer [112] who introduced the concept of oscillation function modeling theory. As described in [112], if the v component represents texture, then v belongs to the Banach space G with the following definition:

$$G = \{v \mid v = \nabla \cdot (\vec{g}), \vec{g} = [g_1, g_2]^T, g_1, g_2 \in L^\infty(\Omega)\} \quad (5.7)$$

induced by the G-norm defined as:

$$\|v\|_G = \inf_{\vec{g}=[g_1, g_2]^T} \{\|\sqrt{g_1^2 + g_2^2}\|_{L^\infty} \mid v = \nabla \cdot (\vec{g}), |\vec{g}| = \sqrt{g_1^2 + g_2^2}\}. \quad (5.8)$$

Meyer's decomposition functional is expressed as

$$E(u) = \|v\|_G + \alpha \int_{\Omega} |\nabla u| dx, \quad f = u + v. \quad (5.9)$$

But the fact is that above G-norm (5.8) in Meyer's decomposition functional (5.9) would cause difficulties in numerical implementation, which was then solved by [172] approximating the L^∞ norm of $|\vec{g}|$ by the L^p norm with the proposal of (5.6). The approximation is stated as follows,

$$\|\sqrt{g_1^2 + g_2^2}\|_{L^\infty} = \lim_{p \rightarrow \infty} \|\sqrt{g_1^2 + g_2^2}\|_{L^p}.$$

According to the authors' discussion in their paper, they found out that similar results could be obtained when p was set as: $1 \leq p \leq 10$, while $p = 1$ yielded faster calculations.

The edge weighted second order (EWSO) model

Recently above first order VO model has been developed into high-order. The edge weighted second order (EWSO) model [54] adopted the specific BH regularizer integrating an edge diffusivity function, with excellent decomposition performance. In our work, the loss function is also based on high-order VO model, thus we choose this model for comparison. EWSO is stated in the following form:

$$\begin{cases} E(u, \vec{g}) = \frac{1}{2} \int_{\Omega} (u + \nabla \cdot \vec{g} - f)^2 dx + \alpha \int_{\Omega} \xi(|\nabla f|) |\nabla^2 u| dx + \beta \int_{\Omega} |\vec{g}| dx, \\ \text{with } \xi(|\nabla f|) = 1 - \exp\left(\frac{-C_h}{(\frac{1}{\lambda} |\nabla f|)^h}\right). \end{cases} \quad (5.10)$$

$\nabla^2 u$ in (5.10) is the BH regularizer defined in (5.2), $\xi(\cdot)$ is the edge diffusivity function: h determines how fast $\xi(\cdot)$ changes; λ controls the smoothness of $\xi(\cdot)$; C_h is a constant obtained by computing $\frac{\partial \Phi(s)}{\partial s} \Big|_{s=\lambda} = 0$ with $\Phi(s) = s\xi(s)$. EWSO model has produced higher performance compared with several classic variational models for decomposition (VO model [172] and the infimal-convolution (INFCON) model [31]) and denoising (TV model [140], TGV model [18] and the combined first and second order (CFS) model [125]). The importance of the edge diffusivity function used as the weight matrix was discussed in detail as well.

5.2.2 Convolutional Neural Network (CNN) based Approaches

Deep Image Prior (DIP)

The DIP architecture [170] was a CNN based generator network [65] with randomly initialized input. Its main idea was derived from the well-known U-net [137]. The obvious advantages of DIP lie in: the network itself can produce desired clear images using only the original degraded images without requiring any handcrafted prior; demonstrating the significance of the use of inductive bias; acting as a "Swiss knife" with diverse applications.

Here the optimization associated with our work is taken into consideration

$$\begin{cases} \theta^* = \arg \min_{\theta} \Psi(DIP_{\theta}(z), f) \\ \text{such that } y^* = DIP_{\theta^*}(z) \end{cases}, \quad (5.11)$$

where z refers to the random input that will be mapped to the given degraded image f by the DIP network, θ are the network parameters. y^* represents the final output of the network and θ^* are the optimized parameters through obtaining the minimization of the loss function $\Psi(DIP_{\theta}(z), f)$. A graphic explanation for DIP is presented below.

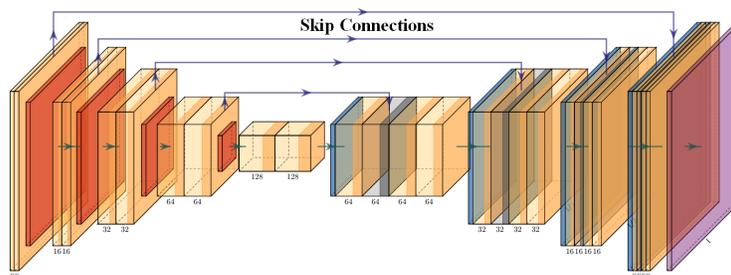


Figure 5-1: Basic architecture of DIP.

As described in [170], DIP was used for focusing on the standard inverse problems of denoising, superresolution, and inpainting. For other particular tasks such as DIP based deblurring [98]. It should be noticed that all the formulations of loss function $\Psi(\cdot)$ are consistent with the fitting terms contained in the variational energy functionals for related image tasks. In addition, DIP serves as an image generator $x = DIP_{\theta}(z)$ mapping a random initialization z to an image x , analogously, the variables defined in variational methods are calculated in the image domain, it thus creates possibility for DIP to produce one certain variable. These properties give us strong inspiration about the possibility for an extension of the DIP by embedding it into variational framework. Please note that the forms of $\Psi(DIP_{\theta}(z))$ presented in Table 5.1(i)-(ii), where u should be replaced by $DIP_{\theta}(z)$ when applying DIP into variational framework.

However, only one DIP is not sufficient for the cases including several variables. For example, TGV functional in (5.3) contains multiple variables: u and $\vec{p} = [p_1, p_2]^T$, which requires two more generators to produce p_1 and p_2 if TGV is the loss function. The research work presented in [58] motivates us with regard to the combination of multiple DIPs.

Double-DIP

On the basis of the single DIP network, it is observed from [58] that multiple DIPs (also called "double-DIP" even though there could be more than two) can successfully separate the image into its natural "simpler" components that are independent of each other. Each component needs to be generated by a DIP and the recombination of these components should reconstruct the original image as much as possible. The general loss function described in [58] is

$$Loss = Loss_{Reconst} + \alpha \cdot Loss_{Excl} + \beta \cdot Loss_{Reg}.$$

note that :

$$\begin{cases} Loss_{Reconst} = |f - \hat{I}| \\ Loss_{Excl} : \text{minimizes the correlation between the gradients of } y_1 \text{ and } y_2 \\ Loss_{Reg} : \text{a task-specific mask regularization} \end{cases} \quad (5.12)$$

\hat{I} denotes the reconstructed image that should be close to the original image f . As an instance of two components $\{y_1, y_2\}$ and a weight mask m , $\hat{I} = m \cdot y_1 + (1 - m) \cdot y_2$. In this way, two DIPs are needed to produce $\{y_1, y_2\}$: $y_i = DIP_{\theta_i}(z_i)$ with randomly initialized input z_i ($i = 1, 2$). In the case of unknown m , one more DIP is further necessary: $m = DIP_{\theta_m}(z_m)$.

Structure-Texture Image Decomposition (STID)

The authors [170] gave another good deep variational example in terms of learning deep variational prior for structure images by using CNNs to replace the TV prior. The multi-scale context aggregation network (CAN) [194] was used to learn the projection operator $\mathcal{P}(\cdot, \Theta)$, then $\mathcal{P}(\cdot, \Theta)$ worked as a modular part to solve the decomposition problem according to the alternating direction method of multiplier (ADMM) in the following formulation:

$$E(u, v; \gamma) = \lambda \mathcal{P}(u, \Theta_\tau) + \int_{\Omega} |v| dx - \gamma^T \int_{\Omega} (f - (u + v)) dx + \frac{\beta}{2} \int_{\Omega} (f - (u + v))^2 dx. \quad (5.13)$$

This is a novel model plugging CNN priors into an iterative smoothing process, where $\beta > 0$ is a penalty parameter and γ is a Lagrange multiplier. The loss function for training $\mathcal{P}(\cdot, \Theta)$

was stated as:

$$Loss_{CAN}(\Theta_\tau) = \sum_{m=1}^M \frac{1}{M} |\mathcal{P}(f^{(m)} + n_\tau^{(m)}, \Theta_\tau) - n_\tau^{(m)}|, \quad (5.14)$$

where n_τ referred to the zero-mean Gaussian noise of variance τ and M (set to 1.5×10^5) denoted the total number of training patches. $\mathcal{P}(\cdot, \Theta)$ is trained on several $\sqrt{\tau}$ ranging from $[45 : -1 : 1]$. After accomplishing the image decomposition by training the projection operator $\mathcal{P}(\cdot, \Theta)$, then each variable u , v and γ in (5.13) are calculated alternatively following the ADMM.

STID demonstrated the feasibility and benefit of using CNN to produce the specific prior in the variational framework. There are three main differences between STID and our work proposed in this paper: 1) no need to pre-train a CNN architecture (CAN in STID), and actually there is totally no pre-training process contained in our work; 2) any type of fast algorithm including ADMM (used in STID) widely used for solving variational functionals is not necessary our work; 3) we explore the potential for simultaneous implementations.

The motivation of our work is to treat the variational multi-task image processing as a unsupervised deep learning problem, in which the unified model is still working as minimization of the loss function. The framework is totally changed since we adopt DIP architecture to generate the final output of each variable with random initialization. Central idea of double-DIP creates the possibility for us to handle the situation that there exist multiple variables contained in the unified model. Our work will be elaborated in the next section.

5.3 Proposed Multi-task Model

5.3.1 Loss Function Inspired by Multi-Task Variational Functional

Motivated by the research [118] developing the VO model to restore blurred images with missing parts and [21] using a unified variational framework to generalize low- and high-order regularization, we first attempt to design novel variational formulations tackling different issues including denoising, inpainting and decomposition in consideration of the advantages of high-order terms and sparsity promotion of ℓ_1 -penalty. In this way, we can fulfill multiple tasks with the general deep variational (DV) model. Without loss of generality, the general variational model can be expressed as the minimization of the following decomposition

functional

$$\left\{ \begin{array}{l} E(u, \vec{p}, \vec{g}) = \frac{1}{2} \int_{\Omega} \Lambda(u, \vec{g}) dx + \int_{\Omega} \zeta(u) dx + \gamma \int_{\Omega} |\vec{g}| dx \\ \text{with } \Lambda(u, \vec{g}) = \begin{cases} (f - u - \lambda \nabla \cdot \vec{g})^2 & \text{for denoising} \\ (m \cdot (f - u - \lambda \nabla \cdot \vec{g}))^2 & \text{for inpainting} \end{cases} \end{array} \right. , \quad (5.15)$$

where the potential function for specific image processing task $\Lambda(u, \vec{g})$ is provided according to the definition of $\Psi(u)$, m is the mask function denoting the broken region, both of them can be found in Section 5.2.1. $\zeta(u)$ represents all the high-order terms in Table 5.1. In (5.15), the corresponding expressions are presented as ℓ_1 -based $\zeta(u)$ as follows

$$\zeta(u) = \begin{cases} TL : & \alpha |\partial_x \partial_x u + \partial_y \partial_y u| \\ BH : & \alpha (|\partial_x \partial_x u| + |\partial_y \partial_x u| + |\partial_x \partial_y u| + |\partial_y \partial_y u|) \\ TGV : & \alpha (|\partial_x u - p_1| + |\partial_y u - p_2|) + \beta (|\partial_x p_1| + |\frac{\partial_y p_1 + \partial_x p_2}{2}| + |\frac{\partial_y p_1 + \partial_x p_2}{2}| + |\partial_y p_2|) \\ EE : & (\alpha + \beta |\partial_x (\frac{\partial_x u}{\sqrt{(\partial_x u)^2 + (\partial_y u)^2}}) + \partial_y (\frac{\partial_y u}{\sqrt{(\partial_x u)^2 + (\partial_y u)^2}})|) (|\partial_x u| + |\partial_y u|) \\ TC : & \alpha |\partial_x (\frac{\partial_x u}{\sqrt{(\partial_x u)^2 + (\partial_y u)^2}}) + \partial_y (\frac{\partial_y u}{\sqrt{(\partial_x u)^2 + (\partial_y u)^2}})| \end{cases} . \quad (5.16)$$

In the context of variational methods, some fast algorithm [160, 162, 164, 165, 211] is usually designed for the minimization of the energy functional formulated in (5.15) with consideration of simplification and effectiveness of the algorithm. There are three main computational difficulties contained in this classical approach: (i) The main critical issue for solving (5.15) is the inevitable high order derivatives in numerical implementation, which is tedious and prone to errors; (ii) Alternating directional optimization strategy based fast algorithm is one solution, but the number of parameters (which can be considered as hyper-parameters in loss function in deep learning approaches) needed to be manually tuned will increase when auxiliary variables are introduced; (iii) Parameter sensitivity. In particular, inspired by CNN based unsupervised learning approaches [58, 170], we propose to employ the classical variational functional to manage the CNN modules, and the optimization algorithm commonly used for solving the variational functional is not necessary any more. Therefore, the computational difficulties discussed above can be avoided. One of the most significant points of our work is hence to show that the mutual benefit is indeed realized when a useful

connection is established between these two approaches.

5.3.2 Multi-Task Deep Variational Architectures

Reduced Version (RV): Image Restoration including Denoising and Inpainting

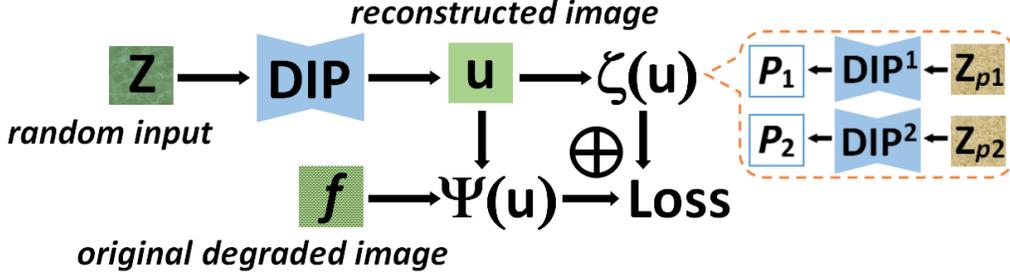


Figure 5-2: Reduced version of the framework without decomposition capability. For TGV, the module in dashed box need to be included into ℓ_1 -based $\zeta(u)$. For other models, it should be omitted.

It can be easily observed that when $\gamma = 0$ and $\lambda = 0$, (5.15) will reduce to the pure high-order based functional without the capability of decomposition. As in the models of TL, BH, EE, and TC, there is only one variable u that needs to be calculated, just one CNN architecture is involved in the framework to realize this reduced version. The associated CNN architecture used in our model is the DIP module from [170]. For the reduced version in relation to TGV model with more than one variable to be obtained, we can choose the idea of coupled DIPs inspired by [58], which need another two CNN generators to gain $\vec{p} = [p_1, p_2]^T$. In order to demonstrate the precise incorporation procedure, we derive the loss functions in reduced version (5.17) and full version (5.18) for our proposed deep variational model respectively and illustrate them specifically in Fig. 5-2 and 5-3 subsequently. With the introduction of CNN based architecture and the condition $\gamma = 0$ and $\lambda = 0$, the original minimization problems of energy functional (5.15) are transformed into the optimization problems of the following loss functions,

for TL, BH, EE and TC :

$$\left\{ \begin{array}{l} \mathbf{Loss}_{RV} = \frac{1}{2} \int_{\Omega} \Psi(DIP_{\theta}(z)) dx + \int_{\Omega} \zeta(DIP_{\theta}(z)) dx \\ \text{such that } u = DIP_{\theta}(z) \end{array} \right. , \quad (5.17)$$

for TGV :

$$\left\{ \begin{array}{l} \mathbf{Loss}_{RV} = \frac{1}{2} \int_{\Omega} \Psi(DIP_{\theta}(z)) dx + \int_{\Omega} \zeta \left(DIP_{\theta}(z), \begin{bmatrix} DIP_{\theta_1}^1(z_{p_1}) \\ DIP_{\theta_2}^2(z_{p_2}) \end{bmatrix} \right) dx \\ \text{such that } u = DIP_{\theta}(z), p_1 = DIP_{\theta_1}^1(z_{p_1}), p_2 = DIP_{\theta_2}^2(z_{p_2}) \end{array} \right. .$$

$\Psi(u)$ and $\zeta(u)$ are provided in Table 5.1 and Equation (5.16) respectively. The framework for this reduced version is given in Fig. 5-2. From Fig. 5-2, it can be observed that z is the randomly initialized input (uniform noise) of the DIP architecture and u is the reconstructed image with the same size as the original image f degraded by noise or damage. For situation of multiple variables, more DIPs are plugged into the framework to produce the corresponding variables (e.g. $\vec{p} = [p_1, p_2]^T$ as shown in the dashed box). All the architectures as well as their output layers (explicitly, u produced by one DIP architecture, p_1, p_2 produced by another two DIPs denoted as DIP^1 and DIP^2 in Fig. 5-2) are coupled by the loss function along some laws derived from traditional variational models from (5.1) to (5.5). In this proposed RV, only one DIP considered when applying TL, BH, EE, and TC and three DIPs involved for the case of TGV. All the architectures are trained simultaneously until we obtain the desired output u .

Full Version (FV): Image Decomposition with effects of Denoising and Inpainting

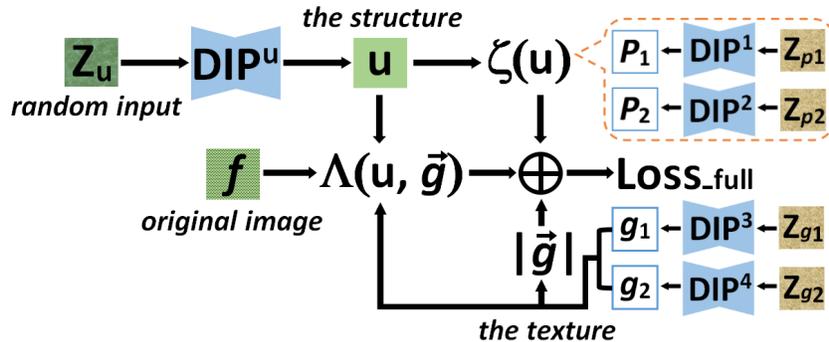


Figure 5-3: Full version of the framework. For TGV-type, the module in dashed box need to be included into ℓ_1 -based $\zeta(u)$. For other models, it should be omitted.

Accordingly, full variational version of (5.15) is formulated when γ and λ are set as positive parameters (the textural components \vec{g} are involved in the loss function in this way). With the introduction of the textural components $\vec{g} = [g_1, g_2]^T$, two DIP architectures are needed to generate these two corresponding variables. Similar with the loss functions described in the reduced version, the models of TL, BH, EE, and TC in this full version have one variable u that needs to be calculated as the image structure component (e.g. the output) and two variables g_1 and g_2 generated as the textural components. Hence three CNN models should be involved in the framework, and five CNN architectures should be involved for $\zeta(\cdot)$ using TGV. The proposed deep variational loss function is as follows:

for TL, BH, EE and TC :

$$\left\{ \begin{array}{l} \mathbf{Loss}_{FV} = \frac{1}{2} \int_{\Omega} \Lambda \left(DIP_{\theta}(z), \begin{bmatrix} DIP_{\theta_3}^3(z_{g_1}) \\ DIP_{\theta_4}^4(z_{g_2}) \end{bmatrix} \right) dx + \int_{\Omega} \zeta(DIP_{\theta}(z)) dx \\ \quad + \gamma \int_{\Omega} \left| \begin{array}{c} DIP_{\theta_3}^3(z_{g_1}) \\ DIP_{\theta_4}^4(z_{g_2}) \end{array} \right| dx \\ \text{such that } u = DIP_{\theta}(z), g_1 = DIP_{\theta_3}^3(z_{g_1}), g_2 = DIP_{\theta_4}^4(z_{g_2}) \end{array} \right. ,$$

for TGV :

$$\left\{ \begin{array}{l} \mathbf{Loss}_{FV} = \frac{1}{2} \int_{\Omega} \Lambda \left(DIP_{\theta}(z), \begin{bmatrix} DIP_{\theta_3}^3(z_{g_1}) \\ DIP_{\theta_4}^4(z_{g_2}) \end{bmatrix} \right) dx \\ \quad + \int_{\Omega} \zeta \left(DIP_{\theta}(z), \begin{bmatrix} DIP_{\theta_1}^1(z_{p_1}) \\ DIP_{\theta_2}^2(z_{p_2}) \end{bmatrix} \right) dx + \gamma \int_{\Omega} \left| \begin{array}{c} DIP_{\theta_3}^3(z_{g_1}) \\ DIP_{\theta_4}^4(z_{g_2}) \end{array} \right| dx \\ \text{such that } u = DIP_{\theta}(z), p_1 = DIP_{\theta_1}^1(z_{p_1}), p_2 = DIP_{\theta_2}^2(z_{p_2}), \\ \quad g_1 = DIP_{\theta_3}^3(z_{g_1}), g_2 = DIP_{\theta_4}^4(z_{g_2}) \end{array} \right. .$$

(5.18)

The full version of our framework with the parameters $\gamma > 0$ and $\lambda > 0$ (usually $\lambda = 1$) is shown in Fig. 5-3. This version in the more complicated structure scales the application range up for image decomposition and decomposition based texture segmentation. Besides the two necessary DIPs (given in the dashed box) producing TGV variables $\vec{p} = [p_1, p_2]^T$ in this version, two more DIPs are needed to generate the texture functions $\vec{g} = [g_1, g_2]^T$. The specific form of $\Lambda(u, \vec{g})$ is presented in (5.15). In this chapter, we intend to demonstrate and

validate that the high performance and natural properties of deep neural networks can take the theoretical mathematics and applied sciences based variational methods to a new level with better outputs. In addition, the classical variational models and useful regularization terms proposed in variational field can provide useful guidance for deep neural network design and formulate strong constraints between each variable.

The surprising property of the proposed FV is the simultaneous implementation of denoising, inpainting and decomposition. As far as we know, it is the first attempt for deep unsupervised learning to achieve so many tasks at the same time. This is the most significant capability that distinguishes our proposed DV model (5.18) from other multi-task networks.

Developed Version: Texture Segmentation with Multiscale and Elastica Constraints

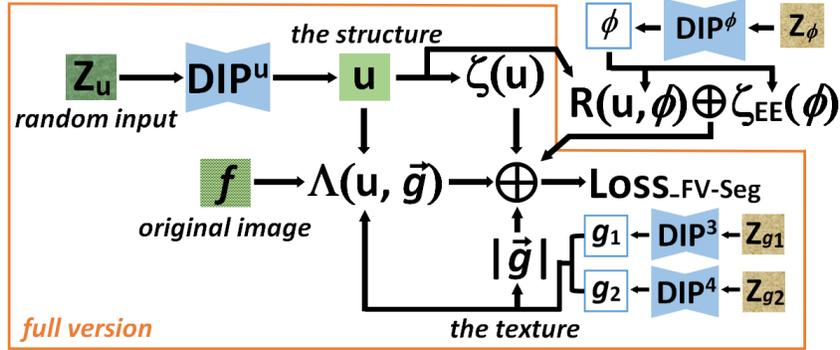


Figure 5-4: Texture Segmentation with Multiscale and Elastica Constraints Framework.

Based on the proposed FV, we can further extend it into image segmentation, which is expected to widen its applications to texture segmentation. In addition, this extension can be better developed by combining with multiscale representation and elastica constraints. In particular, the loss function for this deep variational model is stated as:

$$\begin{aligned}
\mathbf{Loss}_{FV-Seg} &= \int_{\Omega} \zeta_{EE}(DIP_{\theta_{\phi}}(z_{\phi})) dx & (5.19) \\
&+ \int_{\Omega} \Lambda\left(DIP_{\theta}(z), \begin{bmatrix} DIP_{\theta_3}^3(z_{g_1}) \\ DIP_{\theta_4}^4(z_{g_2}) \end{bmatrix}\right) dx + \lambda \int_{\Omega} \zeta(DIP_{\theta}(z)) dx + \gamma \int_{\Omega} \left| \frac{DIP_{\theta_3}^3(z_{g_1})}{DIP_{\theta_4}^4(z_{g_2})} \right| dx \\
&+ \int_{\Omega} (DIP_{\theta}(z) - c_1)^2 DIP_{\theta_{\phi}}(z_{\phi}) dx + \int_{\Omega} (DIP_{\theta}(z) - c_2)^2 (1 - DIP_{\theta_{\phi}}(z_{\phi})) dx \\
&\quad (\text{such that } \phi = DIP_{\theta_{\phi}}(z_{\phi}), u = DIP_{\theta}(z), g_1 = DIP_{\theta_3}^3(z_{g_1}), g_2 = DIP_{\theta_4}^4(z_{g_2}))
\end{aligned}$$

There are three prominent parts in this developed extension. **The first row:** $\zeta_{EE}(\cdot)$ is the EE term of $\zeta(\cdot)$ given in (5.16). The advantages of using EE (also called total absolute curvature [206]) as the regularizer for the mask function ϕ (generated by $DIP_{\theta_\phi}(z_\phi)$) lie in the completion of meaningful missing boundaries, reconstruction of occluded structures of objects and realization of segmentation simultaneously, which are also demonstrated in [162].

The terms on **the second row** are exactly the same as the full version (the first loss function) of our framework (5.18), which is a promising model for image decomposition and restoration. Here the TGV regularizer is not suggested for $\zeta(\cdot)$ since similar performance is obtained from the experimental observation, while more variables will be introduced. According to the similar work [202], TV [140] is adopted to measure the multiscale image u and the scale parameter $\lambda > 0$ controls the scale of u . In contrast, we select $\zeta(\cdot)$ as the multiscale representation, introduce deep prior and establish the novel formulation for wider applications.

The third row: c_1 and c_2 are the pixel intensities of foreground and background of the image structure. This term is for boundary extraction in [202], or serves as the data fidelity term for two-phase segmentation in [162]. Like [58, 162], binary representation for the mask ϕ in (5.19) is continued, the advantage is that we don't need the extra regularization loss used in [58] to realize the binarization of the learned segmentation mask. This term used in our DV model can guarantee the generated ϕ to be binary automatically. Moreover, this term naturally gives the constraints for image foreground and background, thus the extra optimization procedure with the image hints to guarantee fore/back-ground used in [58] is not required. As a result, the efficiency is improved. The framework of (5.19) is summarized in Figure 5-4.

Please note that $R(u, \phi)$ refers to the term given on the third row of (5.19). The proposed full version works as a sub-framework which is marked by orange box. With the introduction of another two terms $R(u, \phi)$ and $\zeta_{EE}(\phi)$ to the proposed loss function in full version, its distinguished property is continued as well as developed: image segmentation with effects of multiscale, structure-texture decomposition and restoration can be achieved simultaneously. The feasibility and potential advantages of our deep variational model are further proved through this extension in this section.

We have successfully achieved the unsupervised learning based deep variational model

for multiple image processing tasks and we will show its effectiveness in next section with extensive experiments for various applications.

5.4 Experimental Results

Platform and Dataset. All the experiments are conducted using Tesla V100 GPU. 30 natural images including both grayscale and colorscale (256×256 and 512×512) are set as our testing images, of which the original color images prepared for decomposition are taken from [45], the alpaca test images used in segmentation experiment are from the Alpaca Molecular Research Team¹ and the others are chosen from public datasets of set12, set14, BSD500 and BM3D². The proposed approach is compared with traditional ones which adopt either CNN or variational methods, and the developed one, TV Regularized DIP, respectively. The same initialized conditions are used for all methods to ensure a fair comparison.

Parameter setting. One important feature of our proposed model (5.18) with its reduced version (5.17) and developed version (5.19) is that the results are robust to the choice of parameters and no more additional parameters will be introduced. For (5.18), $\alpha = 0.001$, $\beta = 0.005$, $r = 0.05$, $\lambda = 1$ are fixed throughout the experiment section. For (5.17), $r = 0$, $\lambda = 0$. For (5.19), the settings keep the same.

5.4.1 Experiments for Denoising

In this subsection, experiment results by BM3D [49], TV [140], DIP [170], DIP-TV [98] and our proposed model (5.15) on grayscale images with different input noise level are shown in Table 5.2. The input signal-to-noise ratio (SNR) ranges from 5dB to 25dB. Comparisons are made with their computed average SNRs to evaluate the denoising performance. It is observed that our proposed model can produce comparable results among all above methods in denoising.

For color images, we first take one typical example to show the ability of our proposed approach for Pepper and Salt noise removal, visual comparison of some selected results is given in Figure 5-5. Some similar results are not contained in this visual comparison

¹Alpaca Molecular Research Team is from School of Pharmacy and Biomedical Sciences of Curtin University. The images were captured for the purposes of alpaca colour genetics research, with the verbal consent of the owners/carers of the animals.

²http://www.cs.tut.fi/~foi/GCF-BM3D/index.html#ref_results

Table 5.2: Average SNR for denoising comparison (grayscale).

Model \ Img Noise	SNR=5 $\sigma=76.26$	SNR=10 $\sigma=53.43$	SNR=15 $\sigma=30.02$	SNR=20 $\sigma=14.24$	SNR=25 $\sigma=5.12$
BM3D	19.20	21.78	24.61	27.42	30.4
TV	17.79	20.20	23.01	26.07	29.14
DIP	18.60	21.35	23.82	26.34	28.49
DIP-TV	19.36	21.89	24.37	26.79	28.89
RV-TL	19.18	21.08	24.34	27.69	31.11
RV-BH	19.07	21.19	23.72	27.53	31.24
RV-TGV	19.03	21.05	24.28	27.48	30.51
RV-EE	18.84	20.96	24.17	27.23	30.54
RV-TC	18.76	21.16	23.80	27.34	30.32

in consideration of paper length. The necessity of using the classic Multi-channel Total Variation (MTV) [189] framework to couple the image information of each channel is then discussed in Table 5.3. Technically, different variational coupling methods refer to the corresponding coupled regularization terms. For color image processing problem, restoring each channel separately using the first order variation (e.g. layered-channel TV (LTV)) models usually results in blurred edges due to different diffusion in each channel. To consider the coupling effects, Sapiro and Ringach [143] proposed an anisotropic filter to diffuse color image according to directions of maximal and minimal changes in an image. The concept of Riemannian geometry and law of Polyakov action in theoretical physics were used by Brook et al. in [22] to their proposed energy minimization models. A celebrated Color TV (CTV) model was proposed in [17] by considering the global coupling of different channels in color image. Most recently, Yang et al. [189] explored the Multi-channel TV (MTV) based only on the local coupling information of different color channels. As described in [189], restoration of similar quality was obtained from both MTV and CTV regularization while MTV had higher computational efficiency than the CTV related models. On the other hand, the common way for deep learning loss function tackling color images is in the form of LTV. In this section, we discuss the necessity of the MTV framework for our proposed deep variational model on color images compared with the LTV framework. Since there are several types of regularization terms, all expressions of them are given in Table 5.3, where $i = 1, 2, \dots, m$ denote the i th channel of a color image.

Figure 5-5 (a) gives the original synthetic image and (b) shows the noisy image, which contains the Pepper and Salt noise with the standard deviation 0.1. Denoising results

Table 5.3: Different regularization terms in frameworks of LTV and MTV

Terms	LTV based	MTV based
TL	$\sum_{i=1}^m \int_{\Omega} \Delta u_i $	$\int_{\Omega} \sqrt{\sum_{i=1}^m \Delta u_i ^2}$
BH	$\sum_{i=1}^m \int_{\Omega} \nabla^2 u_i $	$\int_{\Omega} \sqrt{\sum_{i=1}^m \nabla^2 u_i ^2}$
TGV	$\sum_{i=1}^m \int_{\Omega} \nabla u_i - \vec{p}_i + \sum_{i=1}^m \int_{\Omega} \varepsilon(\vec{p}_i) $	$\int_{\Omega} \sqrt{\sum_{i=1}^m \nabla u_i - \vec{p}_i ^2} + \int_{\Omega} \sqrt{\sum_{i=1}^m \varepsilon(\vec{p}_i) ^2}$
EE	$\sum_{i=1}^m \int_{\Omega} (\alpha + \beta \kappa_i) \nabla u_i $	$\int_{\Omega} \sqrt{\sum_{i=1}^m [(\alpha + \beta \kappa_i) \nabla u_i]^2}$
TC	$\sum_{i=1}^m \int_{\Omega} \kappa_i $	$\int_{\Omega} \sqrt{\sum_{i=1}^m \kappa_i ^2}$

from the Layered Total Curvature(LTC) [165], Multi-channel Total Curvature (MTC) [165], DIP [170] and our proposed models (RV-TC (layered-channel), RV-TC (multi-channel) and FV-EE (multi-channel)) are presented in (c) to (h), respectively. Here only part of models are presented in Figure 5-5 since they are similar in visibility, but their major differences in numerical evaluation (PSNR) are all provided in Table 5.4. It can be observed that CNN based architecture has better capabilities boundary preserving, but our proposed models can further improve this advantage. The popular multi-channel coupling method widely used in variational field still keeps its superior performance when coupling with deep learning technique compared to the layered calculation method, which is reflected in the resulted PSNR presented in Table 5.4. Hence the multi-channel coupling method will be used for all the color cases throughout the whole experiment section.

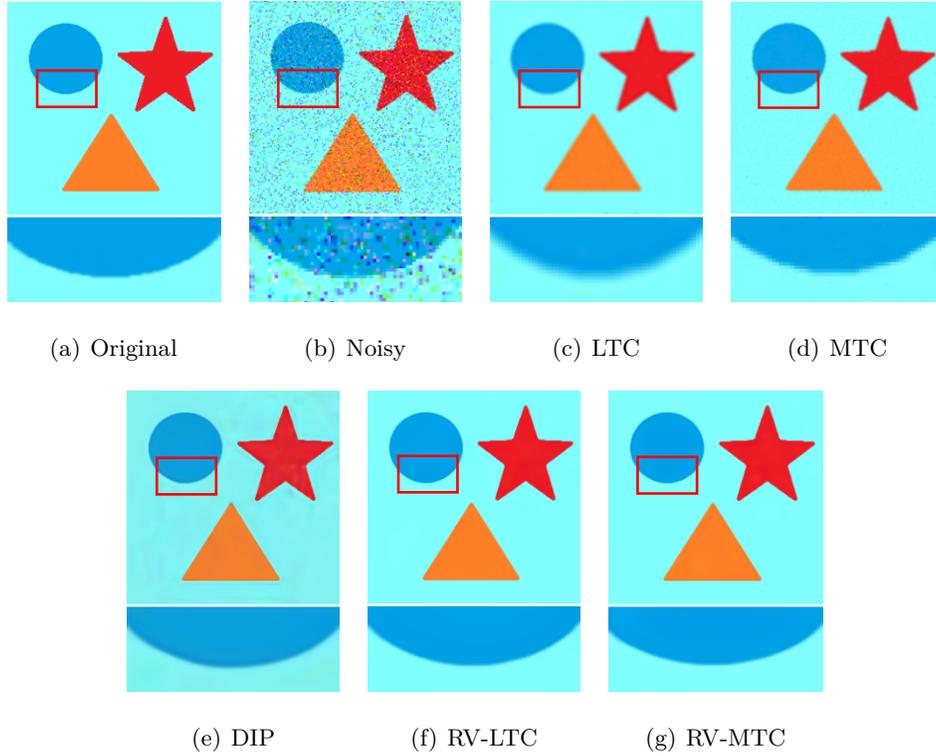


Figure 5-5: Comparison of the effects between LTC [165], MTC [165], DIP [170] and our proposed model. The detail differences are highlighted via rectangles. LTC and MTC results were presented in [165], which are put here directly.

Table 5.4: Comparison of PSNR for Pepper and Salt noise removal (color synthetic), and the difference between coupling methods using layered-channel and multi-channel

Noisy Img and Models (PSNR: dB)						
Noisy	MTV [189]	LTC [165]	MTC [165]	DIP [170]	DIP-LTV [98]	DIP-MTV [98]
13.85	27.07	28.87	29.45	26.86	27.21	27.35
Proposed layered-channel RV (PSNR: dB)						
RV-TL	RV-BH	RV-TGV	RV-EE	RV-TC		
35.03	35.05	35.21	35.33	35.25		
Proposed multi-channel RV (PSNR: dB)						
RV-TL	RV-BH	RV-TGV	RV-EE	RV-TC		
35.66	35.73	35.42	35.71	35.80		
Proposed layered-channel FV (PSNR: dB)						
FV-TL	FV-BH	FV-TGV	FV-EE	FV-TC		
35.11	35.18	35.22	35.26	35.35		
Proposed multi-channel FV (PSNR: dB)						
FV-TL	FV-BH	FV-TGV	FV-EE	FV-TC		
35.57	35.59	35.63	35.69	35.83		

5.4.2 Experiments for Inpainting

In order to demonstrate the performance of inpainting of the proposed architectures, results obtained by pure variational methods (the classic MTV model [189] and MTC model [165]) and pure deep learning approach (original DIP [170]) are used for visual comparison as shown in Figure 5-6. The DIP-TV proposed by [98] targeted image did not implement the application for image inpainting. Thus it is not included in the comparison in this section. In Figure 5-6, (a) shows the original undamaged image. The damaged image is given in (b). The damage type is made complicated and large so as to illustrate that our model can inpaint effectively when tackling such severe contamination. The results by MTV, MTC and DIP are shown in Figure 5-6(c)-(e). Figure 5-6(f) shows the result of our proposed RV-TL. Detailed differences can be observed in the small cropped blocks provided at the bottom of Figure 5-6(a)-(f). It can be seen that all the models can achieve inpainting task well, and the proposed model performs particularly well in inpainting and the results are visually much better. For example, the flower boundary is more clear and the water drop is reconstructed better by our RV-TL model shown in the cropped blocks in blue and green.

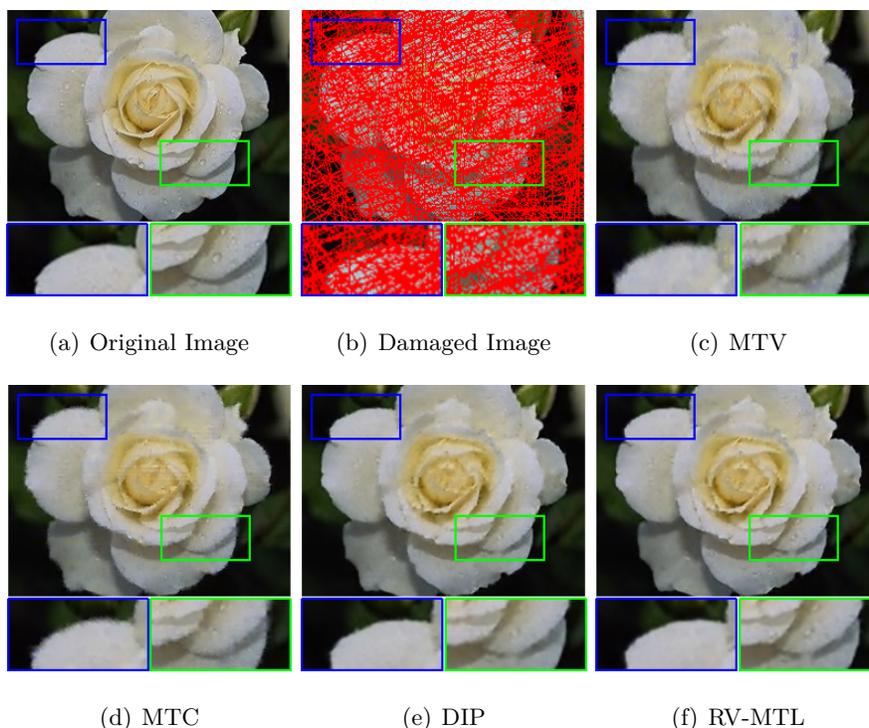


Figure 5-6: Comparison of the effects between MTV [189], MTC [165], DIP [170] and our proposed model (one reduced version and one full version). Some differences are highlighted via rectangles. MTV and MTC results used here were taken from [165].

In Figure 5-7, two experimental examples obtained by [206] and our RV-EE model on inpainting problems are given. These two models have the same target function while the pure variational methods was utilized in [206] with state-of-the-art performance. From left to right, target images, inpainting results obtained by [206] and our RV-EE model are shown in Figure 5-7(a)-(c), respectively. By comparing with the inpainting performance on synthetic and real images of the variational model proposed in [206], it is obvious that geometric features can be much better preserved by our RV-EE model, which is reflected in the detailed water ripple, clear lawn edge and ideally recovered color bars (zoomed regions in rectangles). We can see that deep networks could be a better choice for image priors, promising results can be produced when the same target function is applied.

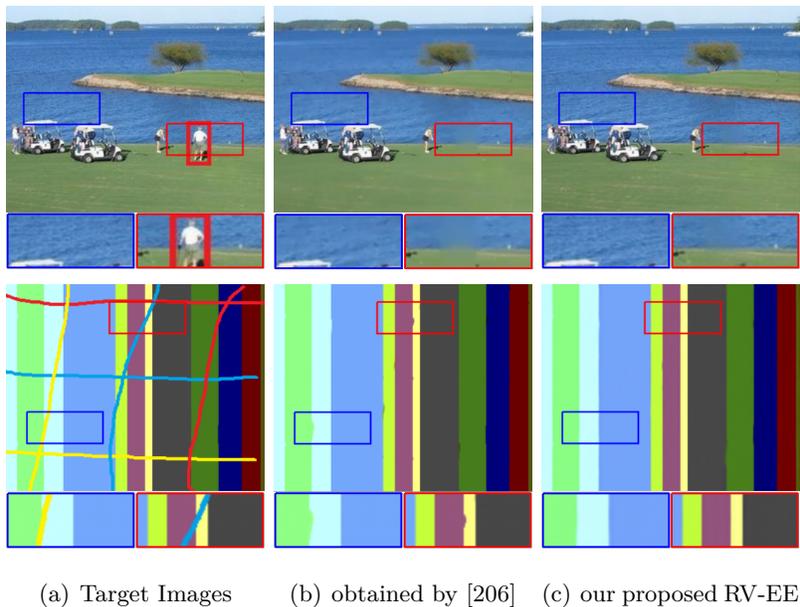


Figure 5-7: Comparison of the effects between [206] and our proposed RV-EE model. Target Images and Results used here were initially presented in [206].

Here the results by our proposed RV-TL and RV-EE are presented in this subsection. For the proposed models in the full version, the stunning feature is that they can realize various tasks (denoising, inpainting and decomposition) at once not only performing well in each task. To the best of our knowledge, they are the first architectures accomplishing such multiple tasks at the same time. Thus we focus on discussing their performance in Section 4.4. Considering both of the simplification and effectiveness, we recommend EE and TC for the potential function $\Psi(u)$ when conducting practical applications.

5.4.3 Experiments for Decomposition

In this section, pure variational EWSO model [54] and deep variational priors [86] are used for comparison in order to show the performance of our proposed model for image decomposition. EWSO model [54] was developed on the basis of the classic model in variational decomposition with excellent results. Here, we choose the proposed FV-TC model as the visual example for grayscale image decomposition. Fig. 5-8 presents the effects of EWSO model with different parameter values α and β (as the authors set for their experiments) and the proposed FV-TC model. Fig. 5-8(a)-(e) give the structure components decomposed from the original texture image by the variational EWSO model and (f) is derived from our FV-TC model. Obviously, (f) can keep more structure information since (a)-(e) all loss a small region information near the face. Meanwhile, the texture is removed more completely in (f), since the headscarf texture is more or less left on the edge near Barbara's hair in other results. More clear differences can be observed by zooming into a small region as shown in Fig. 5-9. It can be seen that our proposed model performs better in edge preserving when extracting image texture, comparing with EWSO model in their best parameter selections.

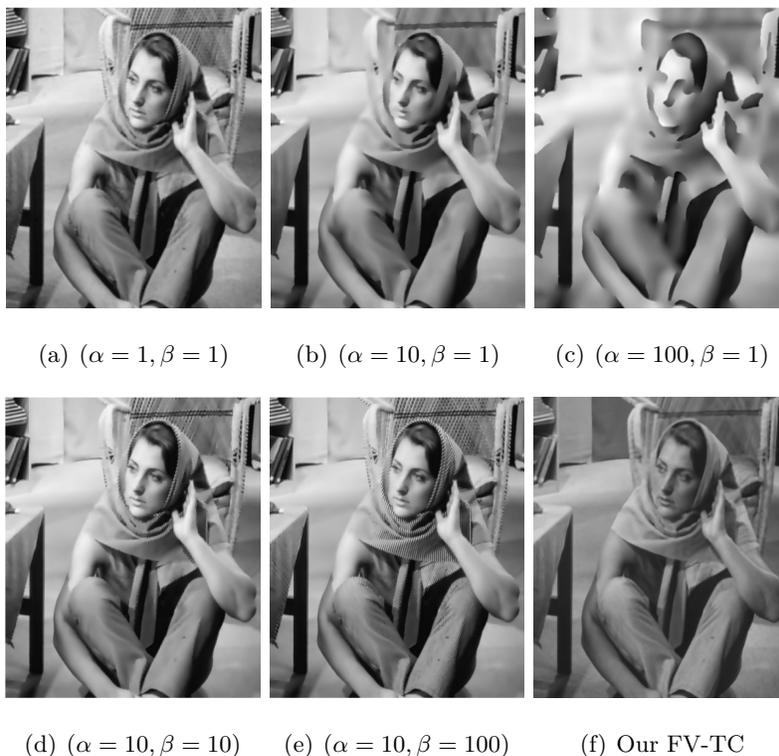
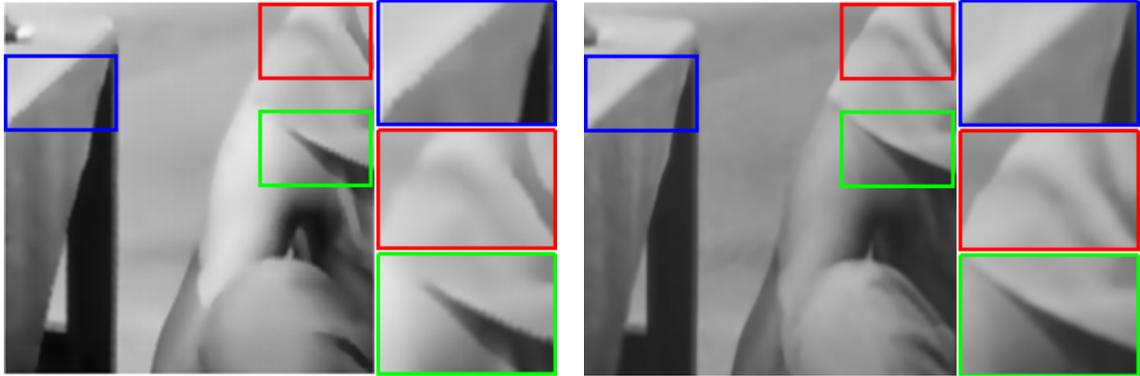


Figure 5-8: Comparison of the image decomposition between EWSO model [54] with various parameter settings (under each image) and our proposed model. (a)-(e) are taken from [54]



(a) EWSO ($\alpha = 15, \beta = 3$)

(b) Ours

Figure 5-9: Zoomed small sub-regions of the results from EWSO model taken from [54] with the best parameter setting and our proposed model for detail comparison. The detailed differences are highlighted via rectangles in three colors.

The approach proposed by Kim et al. [86] is then used to compare with our model (FV-MTL, FV-MBH and FV-MTC) for colors image decomposition. Both STID and our proposed model use CNN-based architecture to produce specific prior, the difference is that there is no pretraining procedure required for the deep network or iterative calculation in our optimization. Our full version for multi-channel case is extended on the basis of [173] by applying the MTV coupling method to the high-order terms $\zeta(u)$ and the texture term $|\vec{g}|$. In Fig. 5-10, (a), (d) and (g) give the original texture images. Results proposed in [86] are presented in Fig. 5-10(b), (e) and (h). The last column including Fig. 5-10(c), (f) and (i) show the results of our proposed approach. From the results in the first two rows, it can be observed that our method achieves higher accuracy reflected in edge and detail preserving. Our decomposed images are closer to the original images, as there are more piece-wise constant regions contained in (b) and (e). In the last row, some texture shade is still left on the vase from [86], which has been highlighted in a yellow rectangle in Fig. 5-10(h). In contrast, our FV-MTC result is smoother and cleaner as shown in Fig. 5-10(i).

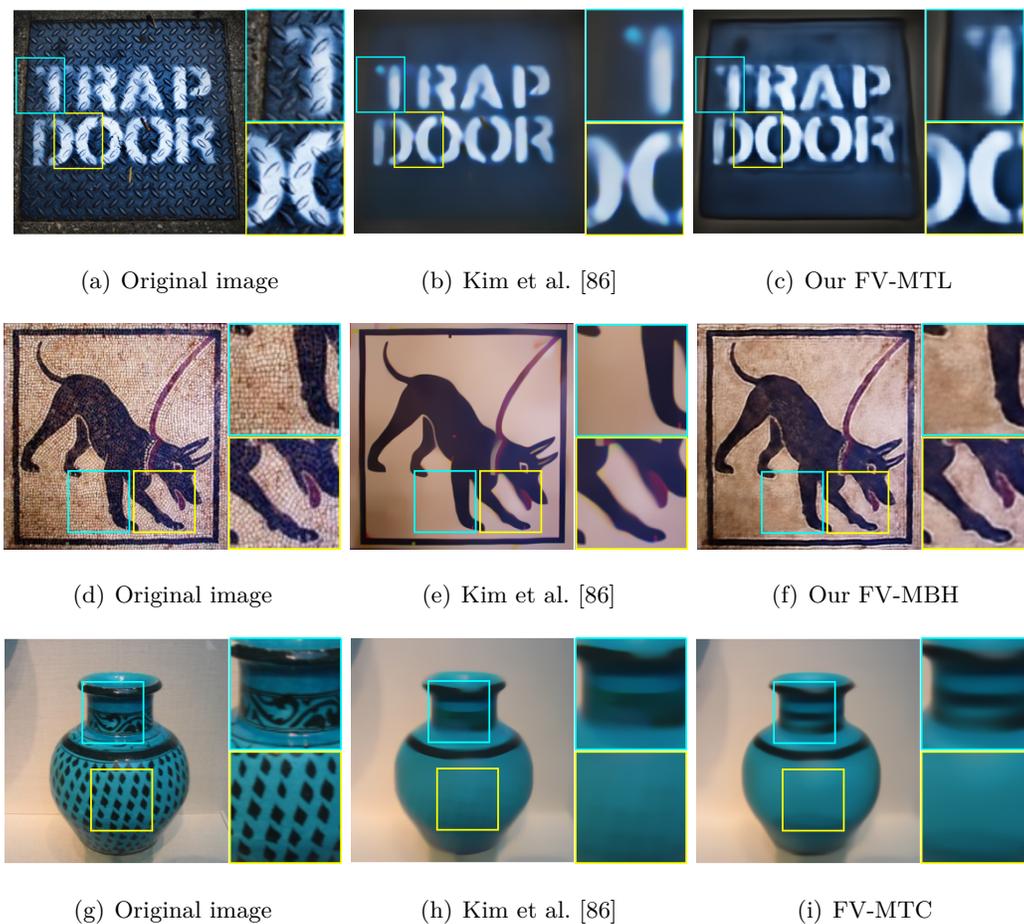


Figure 5-10: Comparison of the effects between Kim et al.'s work [86] and our proposed model (FV-MTL, FV-MBH and FV-MTC) for decomposition in colorscale. (a), (d) and (g) are taken from [45] and (b), (e) and (h) are the results from [86]

5.4.4 Experiments for Simultaneous Implementation of Denoising, Inpainting and Decomposition

In Figure 5-11, the excellent performance of our DV model (full version) on simultaneous implementation of denoising, inpainting and decomposition is shown in (a)-(e). Original images and degraded images (by noises and damages) are given in Figure 5-11(a)-(b). Structure components and texture components extracted through decomposition are displayed in Figure 5-11(c)-(d). From these decomposed results, it can be clearly found that the noises are successfully removed. Meanwhile, the damaged regions are inpainted as well, which demonstrate that both of the structure and texture components can be restored simultaneously by our DV model. Figure 5-11(e) shows the final reconstructed results obtained from our

proposed DV model, combining the structure and texture components together. All these tasks are realized via just one optimization procedure without any pre/post-processing in a totally unsupervised manner. Until now, this outstanding property of our proposed DV model distinguished from other deep networks with promising performance has been illustrated in this subsection.

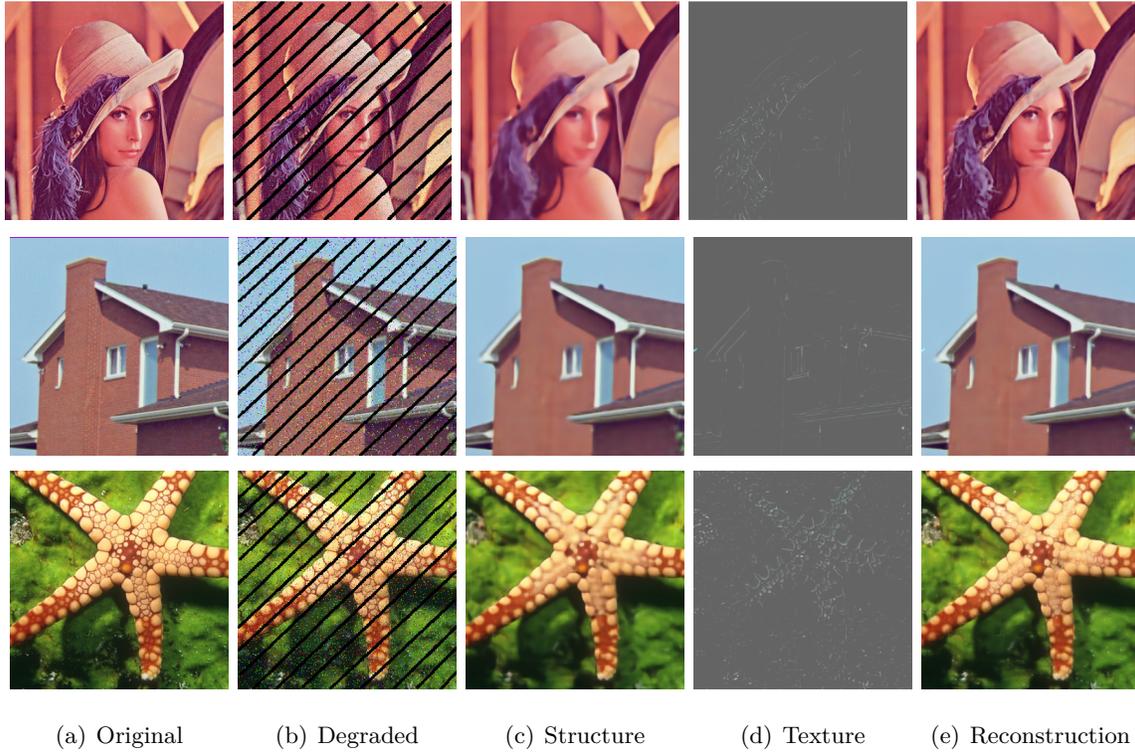


Figure 5-11: Our proposed model (full version) for simultaneous implementation of denoising, inpainting and decomposition in color.

5.4.5 Experiments for Simultaneous Implementation of Image Segmentation with Effects of Multiscale, Structure-Texture Separation and Restoration

In the last experiment, we apply the DV model (5.19) developed from full version for multiscale and object segmentation, with the additional effects of structure-texture extraction and restoration. The variational level set model for multiscale image segmentation [202] and double-DIP networks [58] are used for comparison.

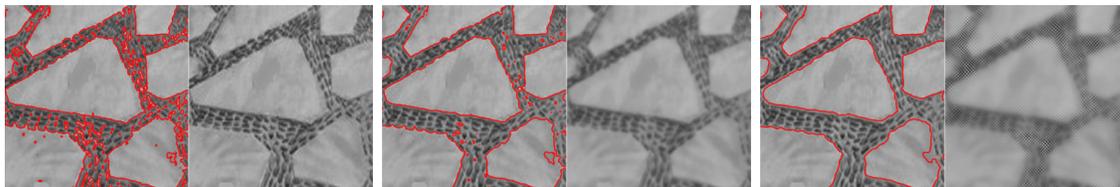
Comparison with pure variational methods

We focus on evaluating our DV model on simple and complex real images in this subsection. The research [202] is the most related and the latest work for multiscale segmentation, which also motivates us a lot. The restoration effect of our model has been discussed in the previous subsections, it will not be described here to avoid repetition. Figure 5-12 shows two types of segmentation results, where the first original image (a) containing textures and the second original image (b) contaminated by Gaussian and salt-and-pepper noises. Figure 5-12(c)-(e) present the multiscale segmentation results of [202] for the simple real image (a). The results by [202] for the complex real image (b) are given in Figure 5-12(i). Compared to [202] for multiscale segmentation on the simple real image, our proposed DV model (5.19) can extract cleaner structure and preserve the boundary better when the scale λ became larger, which can be seen in Figure 5-12(f)-(h). It is hence that the segmentation results with higher accuracy were produced. Experimental results proved that the EE term for ϕ also benefited the segmentation performance, the unexpected impurities can be neglected by our model shown in Figure 5-12(f)-(g). For object segmentation on the complex real image, Figure 5-12(j) showed that our proposed model kept more details of the airplane.



(a) flagstone

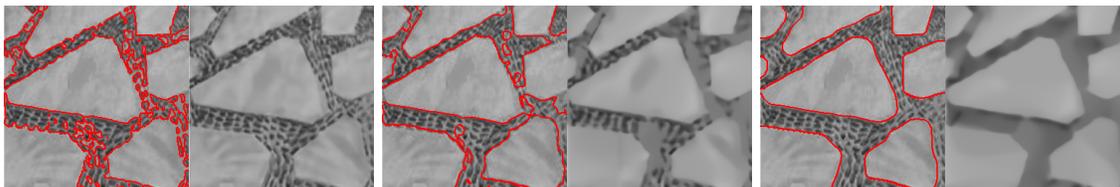
(b) airplane



(c) [202] fine scale ($\lambda = 0.01$)

(d) [202] medium scale($\lambda = 3$)

(e) [202] coarse scale ($\lambda = 10$)



(f) ours, ($\lambda = 0.001$)

(g) ours, ($\lambda = 0.003$)

(h) ours, ($\lambda = 0.01$)

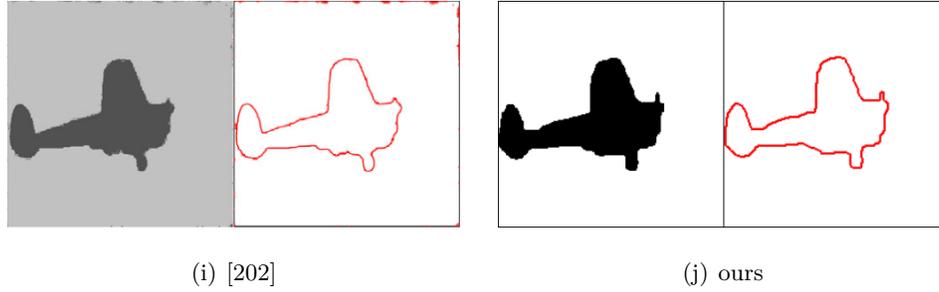


Figure 5-12: Performance comparison between [202] and our proposed model (5.19) for multiscale and object segmentation with textures, impurities or noises. Original images (a)-(b) and Results (c)-(e), (i) presented here were directly taken from [202].

5.4.6 Comparison with double-DIP networks

Visual comparisons of the segmentation results are shown in Figure 5-13. The original images are presented in the first column, some of them containing illumination variation. The segmentation results obtained by double-DIP [58] are shown in the second column, where the final masks come first, followed by the segmentation results plotted with red contour. The third column gives the final segmentation outputs by our work: (c) and (f) provide the decomposed images (extracted structure components), binary final masks and contoured segmentation results from left to right; (i), (l) and (o) show final binary masks and contoured segmentation results respectively. Indeed the complicated background of alpaca images cause difficulty for both the double-DIP and our proposed approach to some extent. However, our approach performs better in higher accuracy and more non-alpaca areas excluded. In addition, hints of foreground and background are needed for double-DIP in advance and added to the loss of the first optimization (around 6000 steps for two optimizations) for stable segmentation. On the contrary, this preprocessing of hint collection is not necessary in our model. Only one optimization with less steps (around 300 steps) is required. Further, double-DIP [58] cannot realize multiscale, decomposition or restoration apart from segmentation in a training. In fact, there are very few deep networks can handle so many tasks like our proposed one does for all we know.

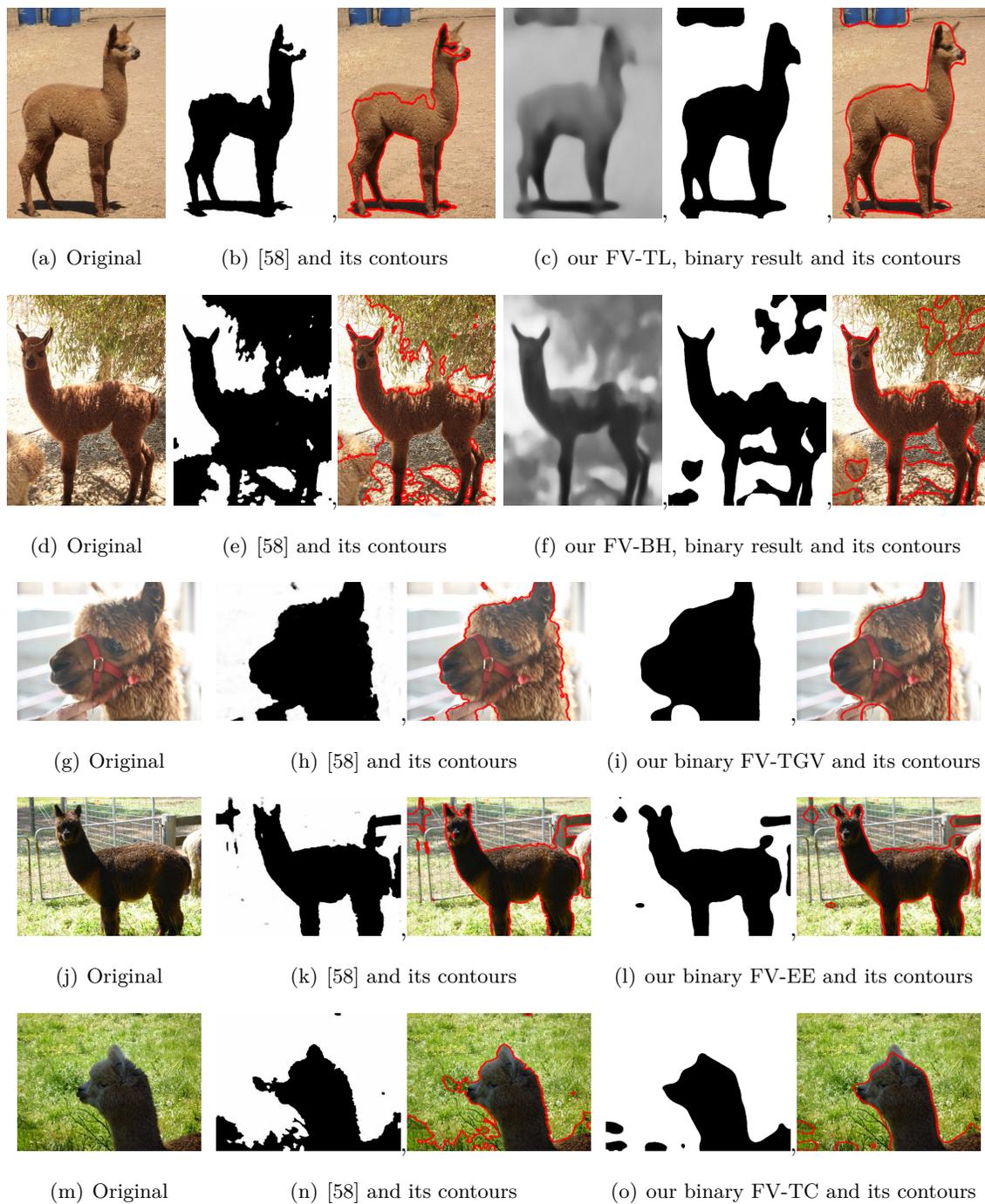


Figure 5-13: Performance comparison between Double-DIP [58] and our proposed model (FV-TL, FV-BH, FV-TGV, FV-EE and FV-TC) for texture segmentation.

5.5 Summary

In this chapter we proposed a general deep variational model (reduced version, full version as well as the extension) via a comprehensive fusion approach. It is able to realize various image tasks in a completely unsupervised way without learning from samples. Technically, it can properly incorporate the CNN based deep image prior (DIP) architecture into the classic variational image processing models. The minimization problem solving strategy is transformed from iteratively minimizing the sub-problem for each variable to automatically minimizing the loss function by learning the generator network parameters. The proposed deep variational (DV) model contributes to the high order image edition and applications such as image restoration, inpainting, decomposition and texture segmentation. Experiments conducted have demonstrated significant advantages of the proposed deep variational model. In the next chapter, we turn into supervised medical image segmentation through another novel deep variational framework, in a totally different fusion manner.

Chapter 6

Deep Variational Learning for Prostate Segmentation

6.1 Introduction

Prostate cancer is one of the common cancers among men. In United States, it still contributes to the highest number of new cancer cases with approximately more than 3 million patients. It is extremely important to detect the presence of cancer cells for early treatment. One reliable way to detect the presence of cancer cells is by extracting a small sample of tissue from the prostate via a biopsy procedure where a needle is inserted through skin to retrieve the tissue. This procedure can only be performed with guidance such as Magnetic Resonance Imaging (MRI). Normally, the location boundary of the prostate region has to be determined by the surgeon manually. This is a time-consuming task and the results might vary among different surgeons. Such subjectivity will lead to discrepancy and longer surgery preparation time.

It is hence essential to develop a framework which allows for efficient and robust prostate segmentation based on medical images. Attempts to automate medical image analysis were reported as early as in 1970s. In the last two decades, people have utilized basic pixel-level features such as edge and curve to establish some basic mathematical shape models for segmentation. The whole framework mainly consisted of manually arranged set of rules.

⁰© 2019 IOS Press. This chapter is based on, as well as with permission from [Tan, L., Liang, A., Li, L., Liu, W., Kang, H., & Chen, C. Automatic Prostate Segmentation Based on Fusion Between Deep Network and Variational Methods. *Journal of X-Ray Science and Technology*, 2019. The final publication is available at IOS Press through <https://doi.org/10.3233/XST-190524>].

They do not adapt well with the inhomogeneity of MRI across various devices. Furthermore, the strong reliance on hyperparameter tuning restricts their application. Due to these limitations, it is essential to develop approaches which are capable of learning and adapting to the great variety of MRI acquisition devices.

The application of automatic/semi-automatic segmentation of prostate in medical images has been extensively studied in the last two decades. Various types of methodologies were employed such as contour-based, region-based, and supervised/unsupervised classification. Some approaches rely on features such as primitive edges [141], Active Shape Model [213], Active Contour Model [198], level sets [164,166,171] and atlas-based [53], etc. Each of these approaches has its own advantages and disadvantages. According to [68], these classical techniques share two common limitations. The first is the necessity of having carefully crafted features or prior information on the images, which limit their capability to perform well in general. The second limitation is in addressing the non-uniformity of intensity level in medical images. Normalization techniques or robust features are required for reliable performance.

In recent years, deep learning network focused on Convolutional Neural Network (CNN) is used with great success in medical image analysis. A huge amount of research was conducted on MRI resulting in significant progress. Among them, CNN-based techniques have made significant achievements for prostate segmentation. For example, Yu et al. [195] proposed 3D convolutional network incorporated with a residual connection to enhance the training propagation of the visual information. Meyer et al. [111] proposed a parallel multi-branched residual CNN to involve triple-perspective axis (axial, coronal, and sagittal) to form prostate boundary. Clark et al. [47] modified an existing CNN model, U-net, by incorporating inception reduction blocks combined with the regularization Drop-out technique to replace typical convolutional blocks.

Unfortunately, experiments conducted on many MRI dataset showed that classical CNN approach could not achieve satisfactory outcomes. In general, a large number of parameters of one deep CNN model are determined relying on a huge number of training images. If the deep learning model is conducted on small medical image datasets, it may easily lead to overfitting. We should pay attention to this limitation of deep network on our small datasets and find some ways to tackle it. From our experiments, it has been observed that CNN based models can perform well on approximating prostate location. However, it is

difficult to keep the final curve close to the prostate boundary. This observation prompted us to explore methods which have good boundary capture capacity. The strong ability of variational methods to distinguish edges especially obscure edges interests us. In the last two decades, there have been many variational models [48,148] proposed to achieve excellent performances in image processing such as medical image analysis, pattern recognition, and video surveillance, etc. With a well-established solid foundation in mathematics and high efficiency, variational methods show significant potential for further development.

Deep learning techniques based on CNN and variational methods based on partial differential equations (PDE) are two excellent branches which have shown strong performance in many applications and made great achievements. It is expected that the combination of these two approaches is able to advance the image segmentation significantly. In 2016, one similar work was conducted by Cha et al. [28], where they proposed a computerized system for bladder segmentation in CT urography (CTU). The approach used a deep-learning convolutional neural network to obtain a likelihood map and then refined by 3D and 2D level sets. Their approach cannot be applied for prostate segmentation since the prostate boundaries in MRI are generally not as clear as the bladder edges in CTU. In some cases, the shapes of the prostate in MRI are actually incomplete due to invisible boundaries and certain internal structures.

Our contributions in this chapter can be summarized in three aspects:

- For CNN part, we propose an idea of global histogram matching. With this approach, we ensure the intensity distribution of the MRI dataset is closer to uniformity and the features can be stabilized. Consequently, more prostate shapes can be captured;
- For the variational part, the location reference of prostate provided by CNN helps prevent the segmentation curve from being attracted into other regions with similar intensity. Incorporated variational stage keeps the final curve close to the real prostate boundary;
- With the fusion via deep network and variational methods, a significant accuracy increase is achieved reflected in the evaluation section.

The rest of this chapter is structured as follows. The next section briefly reviews the related approaches in this field. Afterwards, our proposed approach is presented on Method-

ology section. The experiments conducted with performance evaluation and comparisons are described in Experimental section followed by the summary as the last section.

6.2 Related Work

6.2.1 Learning-based Approaches

Typically, CNN architectures are generally applied for classification tasks with single label output such as holistic image observation (e.g deciding whether the image contains a car). Some recent examples were proposed by Luan et al. [103] and Wang et al. [178]. However, it is not likely to be the case for medical image analysis [96]. For such application, it is essential to include localization elements in the process to extract the regions of interest, which allows further detection of the presence and location of the target object such as particular organs or blood vessels. Further observation is also possible by performing semantic segmentation where each pixel of the image is classified (e.g background/foreground).

Instead of having independent inference for each pixel, it would be wiser to process the image input as a whole and generate output with a similar resolution that indicates the class label for each pixel in one go. Fortunately, this is made possible with encoder and decoder architectures which transform the output resolution close to or even the same as the original size. This technique is also adopted for medical image analysis. One well-known approach is the U-net architecture [138]. It learns the feature by downsampling through a few consecutive convolutional blocks followed by symmetrical upsampling layers in an attempt to recover the resolution, thus shaping the model framework like a character 'U'. The resolution recovery is made possible by the concatenated feature maps from the previous convolutional layers on the same resolution level. This allows prior feature mapping to contribute additional information. Another advantage of U-net is the utilization of data augmentation to compensate for the limited availability of the medical image data which is a very common problem in this research field. Such techniques generate additional training data by applying transformations such as flipping and rotation. Since medical images are commonly acquired in 3D manner such as Computed Tomography (CT) scan and MRI, the idea of segmentation has also been expanded and applied on 3D data. For example, U-net was extended into 3D U-Net [46] which works well on general medical image data if sufficient training data is provided.

We selected one of the robust 3D prostate segmentation CNN models known as V-Net [113] for our proposed framework. It is based on the encoder-decoder-architecture similar to U-net, but designed specifically to segment 3D regions extracted from stacked MRI slices. For the problem of the imbalance between the number of foreground and background voxels, a novel objective function was introduced by Milletari et al. [113] based on the Dice overlap coefficient between the predicted segmentation and the ground truth annotation. Data augmentation using random non-linear transformations and histogram matching was applied to deal with the limited number of training volumes.

Since our experiment was performed on two different datasets, the images come from two distinct intensity distributions, unlike the one used by Milletari et al [113]. In this chapter, a global histogram matching approach is proposed to further improve the performance of V-Net. We perform histogram matching transformation on all MRI (both training and testing) to match the same MRI. This is made possible with the preprocessing phase to normalize the distribution of the data intensity. However, it is not sufficient to cover the whole data distribution between training and testing dataset by using the additional histogram matching [64] utilized by Milletari et al. [113]. In this chapter, a global histogram matching approach is proposed to further improve the performance of V-Net. We perform histogram matching transformation on all MRI (both training and testing) to match the same MRI. This is made possible by the preprocessing phase to normalize the distribution of the data intensity.

6.2.2 Variational Methods

Variational image processing works very well in preserving important image features, dealing with situations of large intensity inhomogeneity etc, and requires less memory in computation. A wide range of reliable numerical schemes for variational models are available, and they can be easily implemented and accelerated. Variational methods were developed from mathematical analysis, which focused on investigating the extreme value of energy functional with respect to some unknown function. Functional refers to a mapping relationship between function set and real number set and can be viewed as the function of functions. A variational problem is to search the extreme value of some specific functional. The solution of the functional extreme value problem is usually determined by minimizing a certain energy functional.

The main advantages of variational methods are: (i) better local adaptability: Functional is established on continuous image model, thus the change of some pixel point value at time t only depends on the infinitesimal neighborhood of this pixel point. It is in this sense there exists infinite local adaptability; (ii) high flexibility: when a basic model has been successfully established, some modifications or extensions can be made on this model to obtain results with improved performance and achieve wider applications. In general, the implementation of such modifications and extensions is straightforward and simple; (iii) low calculation cost: analytical solutions can be obtained, which means existing studies of numerical analysis method can help in computation. Thus variational calculus can be a suitable tool for dealing with large scale problems; (iv) high accuracy and stability: this outstanding property of variational methods is that not only results can show desired performance, but the existence, uniqueness, and stability of their numerical solution can also be proved theoretically.

For algorithms used to calculate the energy functional efficiently, we will apply fast methods. High-order terms in functionals are usually solved by iterative method and some explicit schemes. The convergence would heavily depend on the discretization of the non-linear high-order derivatives in gradient decent equation, which is proved to be tedious and prone to errors. Reliable and efficient algorithms will be developed to accelerate the computation based on the state-of-the-art alternating direction method of multipliers (ADMM) [164, 165, 169], including schemes such as projection method, fast Fourier Transform, soft threshold formula, etc. [164, 165]. Our experiments were conducted on a challenging dataset where the resolution and quality of the MRI are limited.

3D CNN V-Net performed well on other data sets but is not suitable for our data set due to limited resolution and low quality of the image (their testing datasets are up to 512×512 , while ours are 256×256). The strong ability of 2D variational method to distinguish edges especially obscure edges motivates us. This combination approach results in significant performance improvements, as detailed in the experiment section.

6.3 The Proposed Approach

The proposed solution consists of 2 major phases. We propose a framework based on the idea of a multi-phased framework. We choose a 3D CNN model V-Net [113] designed specifically

for volumetric medical images for the first phase. The detected prostate-like result is then passed to the next phase for optimization via variational approach. To demonstrate our proposed solution clearly, the entire approach is presented in a flowchart as follows.

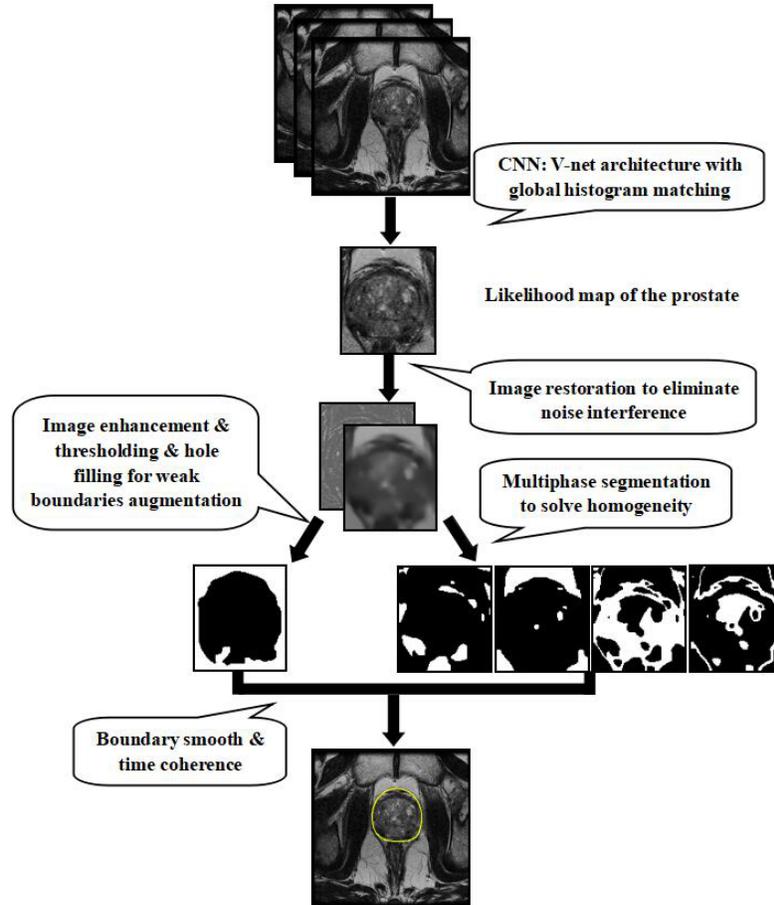


Figure 6-1: The flowchart illustrates how to implement our approach.

6.3.1 V-Net and Training Configuration

Our work is motivated by [113], in which a V-Net approach is proposed for 3D prostate segmentation. There are two main parts in this network structure: the left section involves a compression path and the right part accomplishes the decompression to guarantee that the output will reach the initial size. Convolutions with appropriate padding are all applied, resulting in feature capturing of the input and resolution reduction at the end of each stage. With the objective function based on the Dice coefficient and data augmentation approach, the problems of imbalance between foreground and background and the limited number of

annotated volumes for training are handled well. Our training configuration is quite similar to the one used in [113]. We also utilized the N4ITK bias removal technique [169] on all MRI. The original V-Net already [113] contained one data augmentation preprocessing stage using random non-linear transformations and random histogram matching. Technically, the training images are randomly deformed by using a dense deformation field during every training iteration with random non-linear transformation, and also the intensity distributions of the training volumes in each iteration were changed with random histogram matching. The data augmentation method with these two techniques in the original v-Net can help avoid overfitting caused by small datasets. Our observation reveals that such preprocessing is not sufficient to justify for the whole dataset including both the training and testing data, it may actually lead to loss of important information, especially for image intensity normalization. Furthermore, our MRI images have lower resolution, inconsistent contrast, and unclear prostate boundary in some cases (overall, it has lower quality), intensity normalization is very important. On the other hand, the quality of training and testing dataset of Promise12 dataset is quite consistent which makes the original V-Net easier to produce good results. The disadvantages with only using the original V-Net for our MRI images are demonstrated in our experiments such as the results given in Fig. 6-6(c).

We proposed an approach to solve the data distribution issue. Instead of having training dataset performing histogram matching on multiple random targets, it is more efficient to have an individual global image target for histogram matching. Fig. 6-2 illustrates the effect of our proposed global histogram matching. MRI prostate slices from various patients are shown in the top row, which visualizes the insubstantial and inconsistent intensity level. Instead of performing random histogram matching to cover numerous levels of intensity as proposed in [113], we select a global target for histogram matching (shown here as the middle MRI slice). The first MRI slice from the PROMISE12 [97] training patient case 27 is chosen as the global target. As a result, uniformity of the brightness level on all MRI slices is achieved, improving the stability of the visual features. Results given in Fig. 6-6(d) validate our contribution.

As we are doing segmentation based on 2D medical images and the 3D volumes in our chapter are built from a series of 2D medical images, the histogram matching is usually used between two individual 2D grayscale images. We adopt this approach and achieve significant improvement, the results have been shown in Experiment 6.4.3. Furthermore,

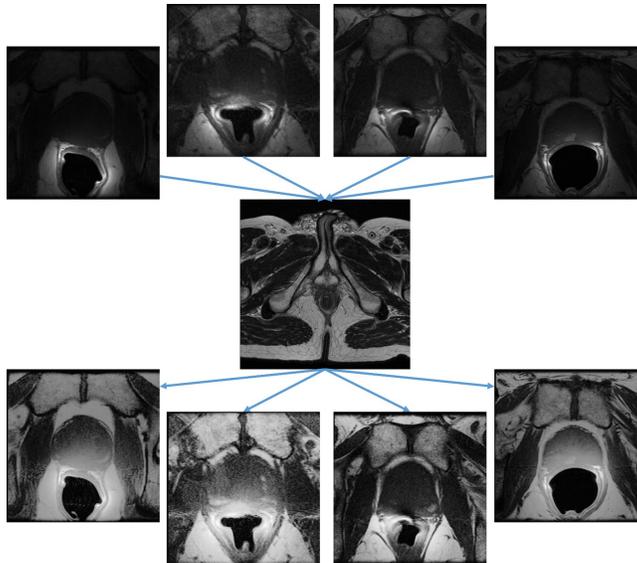


Figure 6-2: Global histogram matching.

Amorim et al. [7] also introduced a 3D histogram matching approach directly transforming the histograms within blocks extracted from the medical data for improving contrast in the context of medical imaging, which will be one of our future research topics.

The improved V-Net shows some improvement in our problem. However, in many situations, it is unable to achieve the desired results. For example, as shown in Fig. 6-5, the segmentation in Fig. 6-5(d) cannot capture the prostate boundaries successfully. It is hence necessary to explore approaches which could achieve this goal.

Dataset arrangement

Availability of medical images is typically limited and restricted due to privacy and confidentiality issue. Prostate MRI dataset is no exception, with only small amount of public database available for experiment purpose. One such database was published in the MICCAI Grand Challenge, PROMISE12 [97]. In this database, 80 real-life patient data are provided (50 for training and 30 for testing). The data acquisition was performed on various devices (vendors) and protocols which makes it even more challenging. We train the V-Net model on the training dataset of PROMISE12 with 50 patients. One MRI slice is selected as the target for global histogram matching to regularize the pixel intensity distribution. Without losing the generality, the first MRI slice from Patient 27 is chosen as the histogram matching target.

Our experiments are conducted purposely on MRI images with a lower resolution where the visual information is limited. The publicly available MRI dataset provided by the Brigham and Womens Hospital [168] are chosen as the testing dataset. It contains MRI data from approximately 230 patients with the resolution of 256×256 per image slice (PROMISE12 dataset can reach up to 512×512). The ground truth of the prostate is available for some patients on the axial perspective of MRI. We selected 35 MRI sets varied in brightness and texture. Some examples are shown in Fig. 6-3.

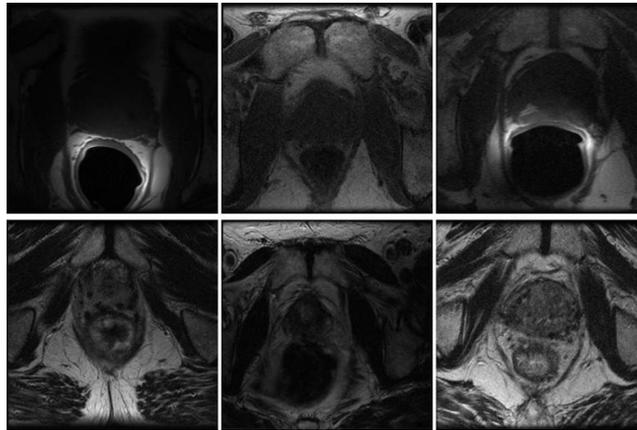


Figure 6-3: Various examples of prostate MRI for testing.

6.3.2 Variational Approach for Prostate Segmentation

Due to the low quality of images and the problem of homogeneity, we need additional aids such as restoration, enhancement before implementing segmentation and boundary smooth. At last time coherence is used for final improvement. The variational method is powerful in such aspects. By incorporating variational methods into the CNN model, object segmentation on low quality images would be improved significantly. In this section, the variational system used in our prostate segmentation will be introduced in detail.

Image restoration via high-order variational model

Medical images like CT or MRI generally contain noises inherently generated by the acquisition procedure. Image restoration [165, 167] can be used for noise removal before other processing. The classic total variation (TV) model [30] is capable of preserving object edges and smoothing flat regions in image restoration. However, it will smear object corners and produce staircase artifacts in the restored images. In order to remedy these side effects,

some researchers have proposed various higher order variational models [165]. In our work, we follow the high order curvature research proposed by Tan et al. [165] due to its excellent performance in denoising and its efficiency. Multi-channel total curvature model is proposed in [165] based on L1 norm for vector-valued image restoration. For our project, it is found out that single channel form is sufficient for gray-scale MRI as follows

$$E(u) = \int_{\Omega} |\kappa| dx + \lambda \int_{\Omega} |f - u| dx \quad (6.1)$$

where $f(x)$ is an observed image defined on the domain Ω , u is the restored image from $f(x)$. κ represents the curvature term given in (2.22).

The first term of (6.1) is the total curvature regularization term, and the second part is data fidelity term based on ℓ_1 norm. The former term is to smooth the images while preserving edges and corners of objects without staircase artifacts. The latter term can preserve image contrast as well as remove outliers and irregularities.

A further enhancement for the weak boundaries is still needed. In some cases, the segmentation curve can be attracted to wrong but strong edges. Hence some image enhancement techniques are desired as presented in the following section.

Multiscale enhancement based on the Hessian matrix

For segmentation, it is necessary to enhance object structures when there exist vague areas or some obscure boundaries. Frangi et al. [57] employed multiscale image enhancement techniques for this purpose. In their research, a Hessian matrix was utilized to determine the likelihood of a vessel. As described by them, an ideal tubular structure in an image generally has a property in the two eigenvalues of its Hessian matrix: λ_1 being very small ($\lambda_1 \approx 0$) and λ_2 is of a large magnitude ($|\lambda_1| \ll |\lambda_2|$). They define a vesselness function via

$$V(s) = \begin{cases} 0 & \text{if } \lambda_2 > 0 \\ \exp(-\frac{\mathfrak{R}_B^2}{2\beta^2})(1 - \exp(-\frac{s^2}{2c^2})) & \text{otherwise} \end{cases}, \quad (6.2)$$

Where β and c are parameters adjusting the line filter to the measures \mathfrak{R}_B and s . \mathfrak{R}_B is given by $\mathfrak{R}_B = \lambda_1/\lambda_2$. The final enhancement filter is determined by different scales: s ,

which is stated as

$$V(\gamma) = \max_{s_{min} \leq s \leq s_{max}} V(s, \gamma) \quad (6.3)$$

where s_{min} and s_{max} refer to the minimum and maximum scales respectively.

After this step, clear boundaries can be obtained for some cases, but homogeneity occurs in other cases. This enhancement approach itself is not sufficient to tackle the problem, which may fail to distinguish the prostate region from the background. Since in many MRI images, it is difficult to distinguish the prostate object from the background. Multiphase segmentation is engaged to divide regions even though there are only small changes in pixel values.

Level sets based multiphase segmentation

In image segmentation, Mumford-Shah model proposed by Mumford D and Shah J [116] is regarded as the most significant region-based model and has been applied to a many applications. In 2001, two-phase Chan-Vese model [40] was proposed for active contours to detect objects in a given image. As the complexity of the images increases, multiphase segmentation is designed to satisfy the demands. Multiphase segmentation models [23] mainly represent different regions by level set functions. In order to reduce the number of level set functions, Chan et al. proposed a multiphase segmentation model [171] which is a generalization of Chan-Vese model. Their scheme can naturally avoid the "overlap" and "leakage" problem. In order to separate an image domain Ω into n subdomains: $\Omega = \bigcup_{i=1}^n \Omega_i$ and $\Omega_i \cap \Omega_j = \emptyset$, [171] defined up to $n = 2^m$ phases and m level set functions. Each pixel belongs to one, and only one phase. A level set function representing an interface $\Gamma(t)$ (t represents time) is defined implicitly as the zero level set of a Lipschitz continuous function $\varphi: R^2 \rightarrow R$ which in turn can be defined in terms of time t as the following signed distance function for efficient computation,

$$\begin{cases} \varphi(x, t) = d(\Gamma(t), x) & \text{if } x \text{ is inside } \Gamma(t) \\ \varphi(x, t) = 0 & \text{if } x \text{ is at } \Gamma(t) \\ \varphi(x, t) = -d(\Gamma(t), x) & \text{if } x \text{ is outside } \Gamma(t) \end{cases}, \quad (6.4)$$

where $d(\Gamma(t), x)$ denotes the shortest Euclidean distance from x to $\Gamma(t)$, from which its Eikonal equation is derived as $|\nabla\varphi(x, t)| = 1$. For $i = 1, 2, \dots, n$, let $(b_{i-1}^1 b_{i-1}^2 \dots b_{i-1}^m)$ be the binary representation of $i - 1$, where $b_{i-1}^k = 0 \vee 1$. The characteristic function $\chi_i(x)$ of Ω_i can be written as the general expression: $\chi_i(x) = \prod_{j=1}^m [b_{i-1}^j + (-1)^{b_{i-1}^j} H(\varphi_j)]$. The energy functional for n phases is obtained as:

$$E(\varphi) = \sum_{j=1}^m \int_{\Omega} \gamma |\nabla H(\varphi_j)| dx + \sum_{i=1}^n \alpha_i \int_{\Omega} Q_i \chi_i dx \quad (6.5)$$

where γ and α_i are positive parameters. The function Q_i is defined as $(c_i - f)^2$, where c_i is a constant which can be obtained by the mean intensity value of f inside Ω_i as: $c_i = \int f \chi_i dx / \int \chi_i dx$.

Boundary smoothing and time coherence for further improvement

After the three steps above, objects with rough and raw boundaries can be obtained. While in some cases, there still exists corners on boundaries and deviation in prostate locations. The boundary smoothing and time coherence techniques are hence employed to improve the final performance. Motivated by total variation [30] for image smoothing, similar method can be extended for boundary smoothing. The final curves produced could be in prostate-like shapes rather than being irregular polygons. Time coherence techniques can be effective in corresponding results with deviation on the basis of center locations obtained by statistics.

The detailed process will be presented in a pseudo code as follows.

6.3.3 Platform

All the experiments are conducted using Matlab7.8 on a PC (Intel (R), CPU: 3.20 GHz, RAM: 16 GB, cores number: 4, architecture: 64-bit). The proposed approach is compared with traditional ones which only used either CNN or variational methods. The same initialized conditions are set for all methods for fair comparison.

6.3.4 Numerical Evaluation

Appropriate metrics for evaluating the performance of image segmentation approaches have been a topic for analysis and discussion. For example, PROMISE12 challenge utilizes metrics such as 95% Hausdorff distance [156] and Dice coefficient [51]. However, there are some

Algorithm: Automatic Prostate Segmentation Fusion via Deep Network and Variational Methods

1. $\{f_1, f_2, f_3, \dots, f_n\}$ are original MRI images from one patient.

$$\{f_1, f_2, f_3, \dots, f_n\} \Rightarrow \boxed{\text{CNN: V-Net integrated with global histogram matching}} \Rightarrow \{I_1, I_2, I_3, \dots, I_n\}$$

2. Initialization: $u_i^0 = I_i$.

$$u_i^{k+1} = \operatorname{argmin} E(u_i) = \int_{\Omega} |\kappa_i| dx + \lambda \int_{\Omega} |f_i - u_i| dx, \quad i = 1, 2, \dots, n.$$

3. (1) $V(\gamma) = \max_{s_{\min} \leq s \leq s_{\max}} V(s, \gamma)$ on u_i^{k+1} .

(2) Initialization: $\varphi^0, f = u_i^{k+1}$.

$$\varphi^{k+1} = \operatorname{argmin} E(\varphi) = \sum_{j=1}^m \int_{\Omega} \gamma |\nabla H(\varphi_j)| dx + \sum_{l=1}^n \alpha_l \int_{\Omega} Q_l \chi_l dx.$$

(3) $\hat{u}_i^{k+1} = V(\gamma) \cap \varphi^{k+1}$.

4. Boundary smooth and time coherence on $\{\hat{u}_1^{k+1}, \hat{u}_2^{k+1}, \hat{u}_3^{k+1}, \dots, \hat{u}_n^{k+1}\}$.

other applicable evaluation frameworks. Extensive review and analysis have been done on various evaluation approaches on 3D medical segmentation. Among these evaluation metrics, we selected seven prevalent ones from the five categories shown in Table 6.1.

Among these methods, ROC (Receiver Operating Characteristic) [70, 157] method is typically used in classification tasks of two classes. To evaluate segmentation performance, we consider the definitions of the ROC curve and AUC (Area Under the Curve) computation [157]. According to their description, AUC is the area of the trapezoid defined by the single measurement point and the lines $\text{TPR} = 0$ and $\text{FPR} = 1$, which is given by

$$\text{AUC} = \frac{\text{TPR} - \text{FPR} + 1}{2} = 1 - \frac{\text{FPR} + \text{FNR}}{2}. \quad (6.6)$$

It is also important to note that the Mean Hausdorff Distance [156] is more stable and robust to outliers compared to the original Hausdorff distance.

Table 6.1: Chosen Evaluation Metrics.

Categories	Metrics
Overlap based	Dice Coefficient [51] & Jaccard Coefficient [77]
Volume based	Volumetric Similarity [24]
Pair counting based	Adjusted Rand Index [76]
Probabilistic based	ROC Curve Area [70, 157]
Spatial distance based	Mean Hausdorff Distance [156] & Mahalanobis Distance [106]

6.4 Experiments

In this section, some numerical experiments on MRI slides of the prostate are presented to validate the performance of our proposed approach. Our experiment results against the 7 metrics introduced in subsection Numerical Evaluation are shown in Table 6.2. They are the average evaluations (along with the corresponding standard deviation) from 35 MRI test cases. We first observe the results from the original variational model and V-Net on the first two columns. As expected, most of the results from various metrics of Variational Model alone are less accurate compared to V-Net model due to the inadequacy of prior information on prostate region. It is fascinating that Variational Model thrives on both ROC area and Mahalanobis Distance metrics, however the application of V-Net and histogram matching together reduces the gap and improves the performance on other metrics (column 3). Most importantly, it can be seen that our proposed fusion approach achieve significant improvement on all evaluation metrics. One example of 3-D reconstruction for

Table 6.2: Seven evaluation metrics on (a) Variational Method, (b) V-Net, (c) V-Net+Histogram matching, (d) The proposed Method. Numbers in bracket represent the standard deviation.

Metrics	Var. Method	V-Net	V-Net+Hist.	Proposed
Dice Coefficient	0.285 (0.165)	0.519 (0.224)	0.581 (0.185)	0.646 (0.138)
Jaccard Coefficient	0.177 (0.120)	0.380 (0.202)	0.432 (0.180)	0.491 (0.147)
Volumetric Similarity	0.436 (0.221)	0.690 (0.275)	0.740 (0.223)	0.831 (0.129)
Adjusted Rand Index	0.225 (0.145)	0.495 (0.215)	0.558 (0.177)	0.620 (0.133)
ROC Curve Area	0.805 (0.123)	0.735 (0.129)	0.758 (0.112)	0.846 (0.088)
Mean Hausdorff Distance	5.732 (2.699)	2.237 (1.908)	1.645 (1.456)	1.380 (1.205)
Mahalanobis Distance	0.782 (0.403)	0.924 (0.605)	0.704 (0.493)	0.684 (0.467)

prostate based on the segmentation from MRI slides is presented below, which serves as an intuitive and visual evaluation. From Fig. 6-4, the advantages of our proposed framework are validated, especially in terms of smoothness, uniformity, and accuracy.

In the following sections, the performance of our approach will be demonstrated visually compared with existing methods. Experiments are conducted on all the images from 35 selected MRI sets, but only selected examples will be shown without loss of generality.

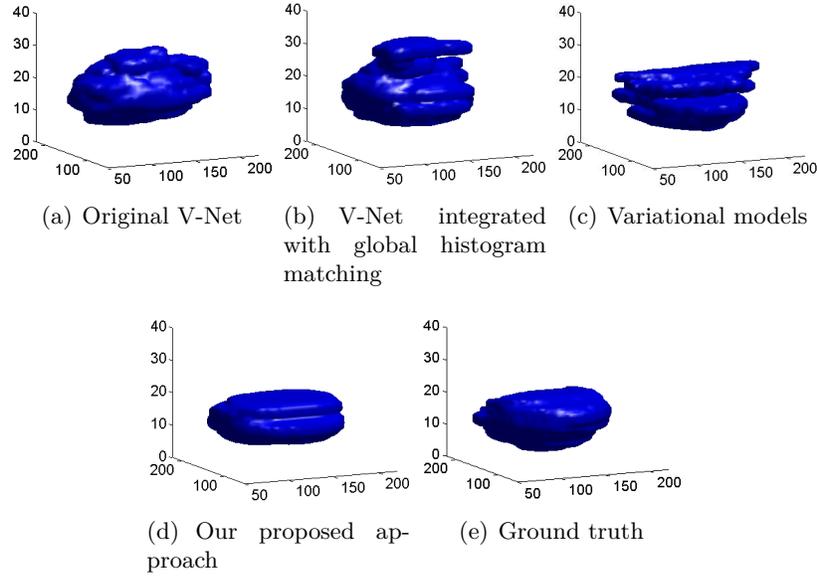


Figure 6-4: 3-D reconstructions on: (a) Original V-Net. (b) V-Net with global histogram matching. (c) Pure variational models. (d) Our proposed framework. (e) Ground truth.

6.4.1 Experiments for Comparison with Pure Variational Methods

All the prostate images of size 256×256 are used as the test images. In these experiments, the results obtained by pure variational methods are used for comparison in order to demonstrate that our fusion approach can achieve better results. Fig. 6-5 shows some results from the variational methods and from the proposed approach respectively. The corresponding ground truth images are shown in red for visual comparisons. Fig. 6-5(a) shows the original images. The results of using pure variational methods (yellow line) and the ground truth (red line) are presented in Fig. 6-5(b). Fig. 6-5(c) gives final results of our proposed fusion method (yellow line) and the ground truth (red line). Zoomed-in images are shown in Fig. 6-5(d) and (e) to better demonstrate the performances. The zoomed views show the region bounded by the blue rectangles in Fig. 6-5(b) and (c).

From the results presented in Fig. 6-5, it can be observed that our fusion method achieved higher accuracy. More non-prostate areas are excluded. Comparisons of local regions clearly demonstrate that our model is capable of improving the segmentation performance.

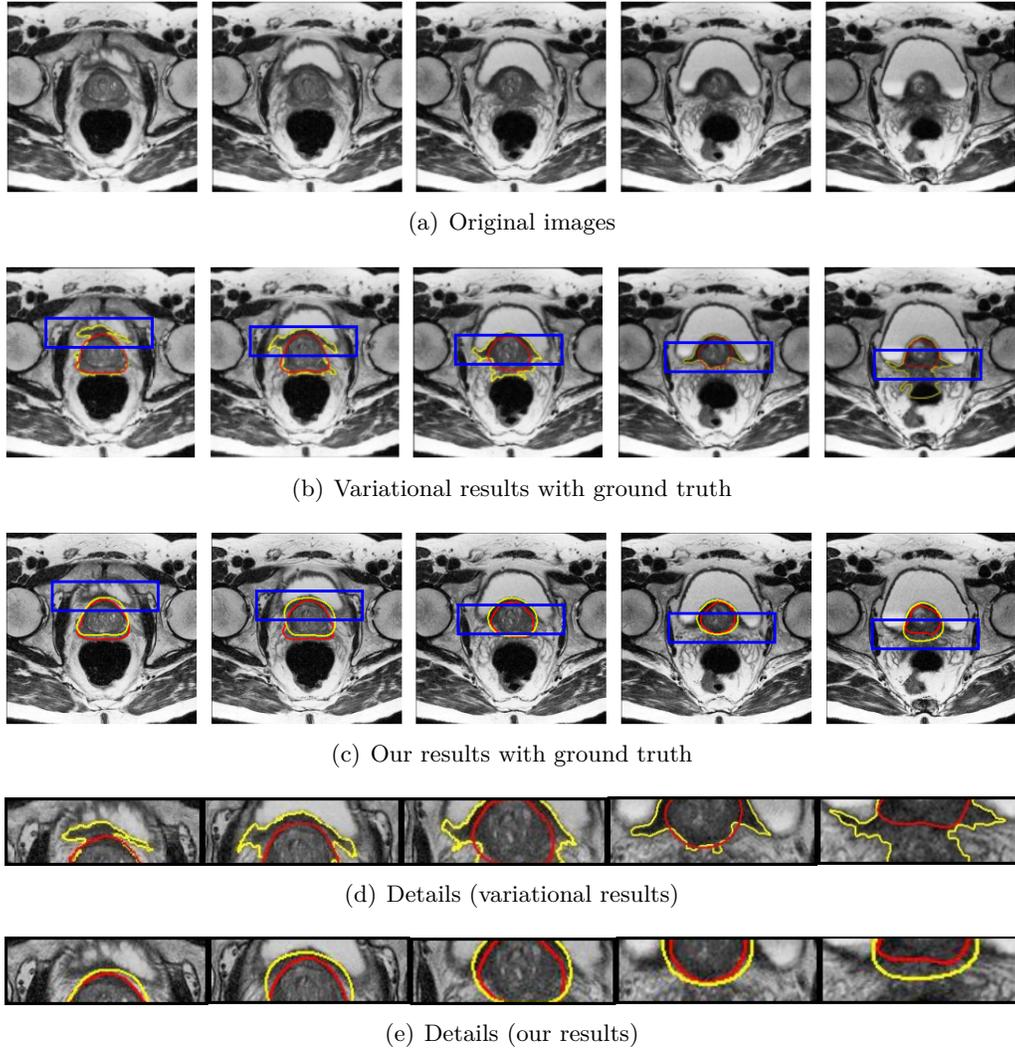


Figure 6-5: The effects of variational models and our proposed approach from patient 6. (a): Original MRI. (b): Variational results. (c): Our results. (d)-(e): Details.

6.4.2 Experiments for Comparison with the Original V-Net

Next, our approach will be compared with the original V-Net. In order to clarify the detail improvements, V-Net integrated with global histogram matching results are also presented in this section. In Fig. 6-6, the first two rows are the original MRI images and the corresponding ground truth. The last 3 rows show results from the original V-Net, V-Net integrated with global histogram matching, and our proposed framework.

Through comparison with the original V-Net, we can see the obvious advantages of our approach, which can capture more prostate objects and reconstruct better prostate shapes from the whole image sequence. Fig. 6-6(d) shows the incorporation with global histogram

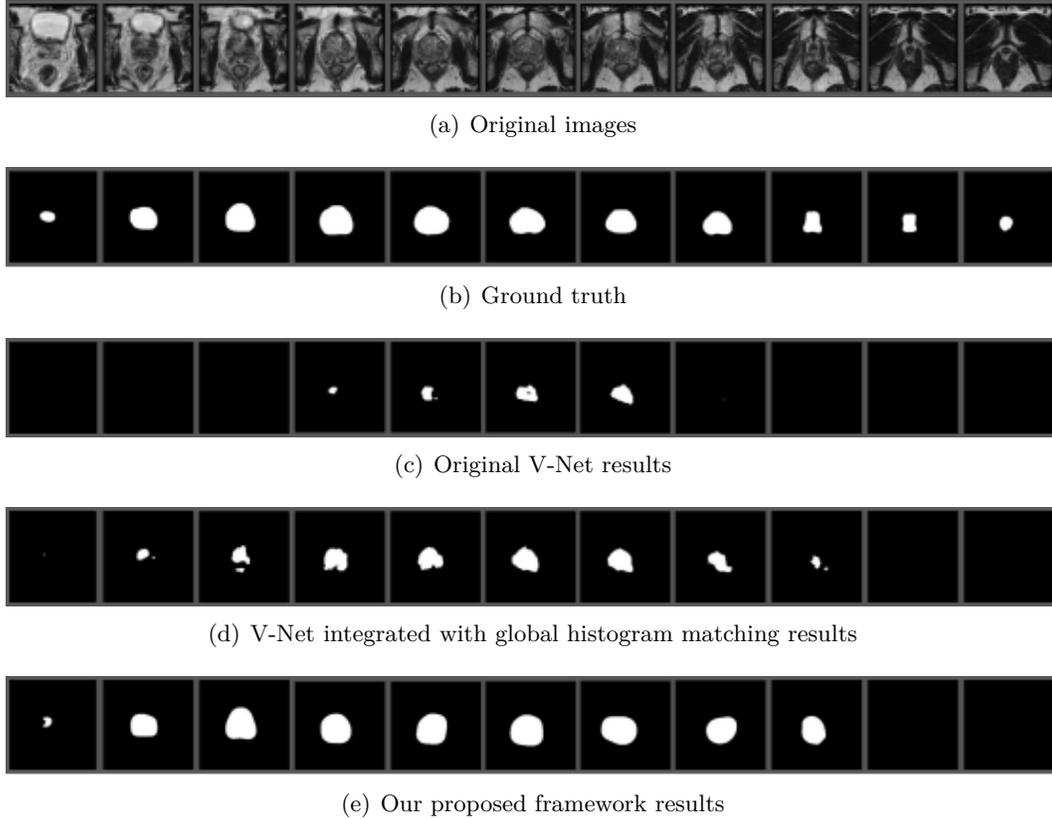


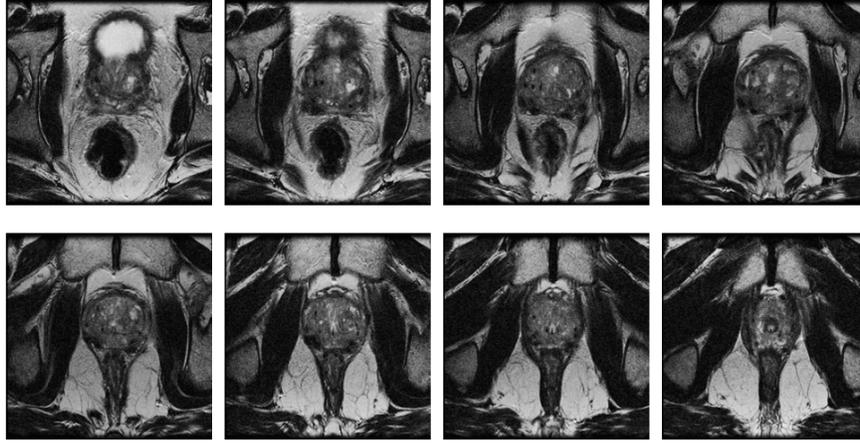
Figure 6-6: Comparison of patient 46. (a): Original MRI. (b): Ground truth. (c): Original V-Net results. (d): V-Net integrated with global histogram matching results. (e): Our proposed complete framework results.

matching can provide more location information of prostate for reference. Then combination with variational methods achieves further improvement as given in Fig. 6-6(e).

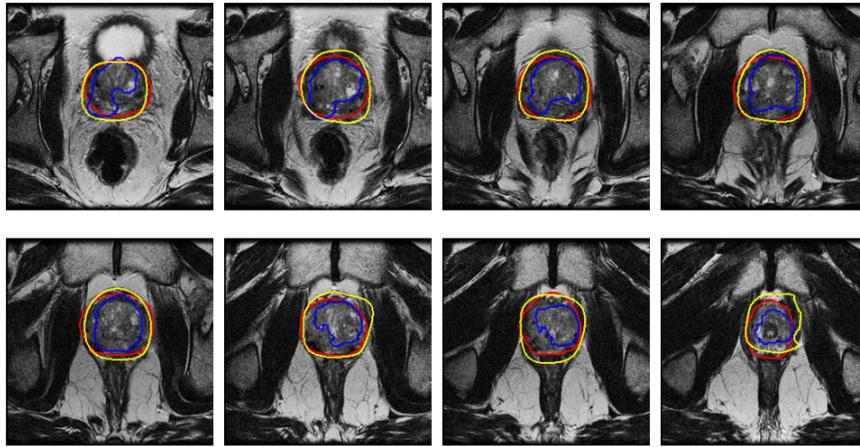
6.4.3 Experiments for Comparison with V-Net Integrated with Global Histogram Matching

The next experiment is to show the improvement of our method over the improved V-Net, specifically, the V-Net integrated with global histogram matching. Fig. 6-7(a) shows the original MRI image. The segmentation results including results from the improved V-Net and our proposed approach are presented in Fig. 6-7(b) with the ground truth shown in red. In Fig. 6-7(b), the blue line shows the result of the improved V-Net and the yellow line shows our results.

It can be clearly seen that our segmentation results are closer to the real boundaries of the prostate, and represent the prostate shapes better. In the first three images in Fig.



(a) Original images



(b) Comparison of segmentation

Figure 6-7: The effects of pure improved V-Net and our proposed approach from patient 29. (a): Original MRI. (b): Results from V-Net integrated with global histogram matching (blue line), our final results (yellow line) and the ground truth (red line).

6-7(b), there exist obvious deformations in the blue curves resulted from the improved V-Net. Clearly the improved V-Net alone cannot reach the same accuracy as our approach, with possible information loss or boundary misjudgment.

6.5 Summary

The focus of this chapter is prostate segmentation for MRI, which is a critical process for guiding prostate puncture and biopsy. Currently, the best results are obtained by Convolutional Neural Network (CNN). However there still exist some problems when we apply CNN for the prostate segmentation, such as data distribution issue caused by insubstantial and

inconsistent intensity levels and vague boundaries of the prostate in MRI. In this chapter, we propose a global histogram matching approach to make intensity distribution of the MRI dataset closer to uniformity. In order to capture the real boundaries and improve the accuracy of segmentation, we employ a module built by variational models to help improve the performance. Extensive experiments are conducted on a data set including different prostate images with limited resolution and quality, and the results validate the proposed approach, with better performance than the existing methods based on the original CNN or pure variational models.

Chapter 7

Conclusion and Future works

7.1 Conclusion

This thesis investigated approaches based on variational methods, stochastic programming and deep learning techniques for pixel-level tasks in computer vision. There are three main objectives.

- The first objective is to develop variational and PDE based methods for various image tasks, including image restoration such as denoising, inpainting for restoring large damages, illusory contour capture and nonlinear based segmentation, as well as exploration of reliable optimization schemes. In chapter 3, we propose the ℓ_1 based multi-channel total curvature (MTC- ℓ_1) model for color image processing. In comparison with the MTV- ℓ_1 model, the proposed model would not smear object corners and yield staircase artifacts. We also compare our proposed model with MTC- ℓ_2 model, and the results highlight the improvement of our model in image contrast preserving. In addition, we observe that the LTC- ℓ_1 would be not able to keep color image edges when applied to each channel respectively, while the MTC- ℓ_1 model can achieve excellent performance in color image edge preservation. In order to improve the efficiency of the proposed model, we develop a fast ADMM algorithm based on an explicit discrete finite difference scheme, which is in fact an intuitive approach for the proposed ℓ_1 optimization problem. We note that the ADMM method performs with higher efficiency than the traditional gradient descent flow method. Numerical experiments validate the performance of the proposed MTC- ℓ_1 model for color image

denoising and inpainting in different cases.

- The existing model based level set method for capturing illusory contours is computational expensive and time consuming. Three major contributions are presented to address this problem. Firstly, we propose a simplified variational level set formulation by taking advantage of the property of signed distance functions. Secondly, we develop the fast ADMM-P method by combining the ADMM method and projection method. Thus, the originally complicated problem is decomposed into a series of simple sub-problems of optimization, and each sub-problem can be easily solved. Lastly, the re-initialization process of the level sets is removed in numerical implementation. Extensive experiments have validated the effectiveness of the proposed ADMM-P approach as it is much faster than the traditional algorithms based on the GDM, which may be due to the merits of the Gauss-Seidel method, FFT method, projection method as well as model reduction. Furthermore, for noisy cases our proposed approach achieves better performance than the original model.
- By using the concepts of nonlinear diffusion and Chan-Vese model, we propose a new general model and apply it into grey and colour image segmentation. Our model can detect regions with different features and has been successfully extended to colour images. Then we propose a new fast algorithm NVPM based on the alternating optimization method and normal vector projection. Also, an accelerated method is applied to optimize NVPM. The good performance of our model and the efficiency of the proposed algorithms has been validated by numerical experiments. Comparisons of results and computational time obtained by ADMM and our proposed approach indicate that NVPM requires shorter runtime while the quality of results is identical.

The second objective is to solve the problem of random noises contained in images when conducting segmentation. Existing variational methods and applications were based on known types of noises. In Chapter 4, this objective is achieved by a fusion approach incorporating stochastic programming theory into variational models. It is also the first work applying stochastic programming into image tasks to determining the uncertainty of noises.

- We propose a novel variational approach for image segmentation with stochastic noises and develop a progressive hedging algorithm to solve them. Our approach has three

advantages: (1) improving segmentation ability for noisy images without the prerequisite that one given model corresponds to one specific noise distribution; (2) realizing meaningful completion of missing boundaries and reconstruction of occluded structures of objects in a highly noisy background; (3) incorporating the ADMM method and a curvature-weighted approach into the calculation procedure to guarantee the segmentation quality on both convergence and efficiency. Extensive experiments were conducted on images with multiple segmentation tasks which are more challenging especially when the image quality is low. Experiment results demonstrate the significant performance improvements of our work. Furthermore, for images with big stochastic noises and damages, our proposed model achieves better performance than the traditional model, which is of great significance for image understanding with problems such as occlusion, large damages or noises.

The third objective is to explore the possibility of fusion approaches motivated by incorporating variational methods and deep learning, with better performance than the existing methods purely from one field. The fusion approaches are elaborated in Chapter 5 and 6 respectively.

- A good example of connecting the pure image processing based variational methods and the well-known CNN based deep-learning architecture is given. We propose a novel deep variational model based on the idea of coupling variational framework with DIPs which can be applied for various types of image tasks without the need for training data. The advantages of our approach are: (1) improving image processing capabilities for images with noises, damage regions, blur distortion and textures; (2) realizing texture segmentation with no need of any prerequisite as a guidance, which successfully extends the applications of image decomposition model and is robust to noises; (3) focusing on model structure dominated performance, no need to design the optimization algorithm that will inevitably introduce extra artificial parameters. For CNN part, classic variational energy functional can provide proper guidance for the architecture parameters tuning and strong constraints for each CNN models effect. For the variational part, introduction of CNN as variable generator can substitute the expensive mathematical calculation with respect to each variable and alleviate the hyper-parameter sensitivity. Extensive experiments were conducted on images with

various applications to demonstrate the encouraging performance of our work which is comparable to and, sometimes greater than the state-of-the-art methods.

- Among the previous classical medical image analysis methods focusing on prostate segmentation techniques, the ones being very close to our research are the pure variational method-based approaches [30, 57, 165, 171] and the deep CNN based v-net model [113]. We proposed a prostate MRI segmentation framework based on the well-known Convolutional Neural Network architecture V-net and the variational method. We sequentially embed global histogram matching and a variational system into the architecture to enhance the segmentation quality on both coverage and smoothness. Compared with the pure variational methods through experiments, the advantages of our proposed fusion approach lie in: i) the likelihood map obtained from deep CNN model can provide some essential prior reference information for the pure variational methods, which makes the final results more stable; ii) the problem of the final curves are easily attracted into the fake boundaries caused by homogeneous intensities is addressed by the global histogram matching. As for the comparison with v-net model on the low quality dataset, our proposed fusion approach can achieve significant improvements in: i) more prostate objects can be detected by applying the global histogram matching; ii) not only the localization elements to extract the regions of interest are recognized more accurately, but also prostate shapes can be better represented with the ability to capture real boundaries. Extensive experiments were conducted on MRI with lower resolution which is more challenging due to the limited image quality. Experiment results demonstrate the significant performance improvements of our proposed approach.

7.2 Future Works

The satisfactory performances have validated all of the proposed approaches, but much room still exists for further improvement. Below we list some possible directions for future works:

- For image processing techniques based on variational methods, the strategies of model simplification and constraint projection can be easily extended to other variational

level set models with/without curvatures. In chapter 3, we concentrate on the foreground reconstruction without restoring the background shape. In the future, we will investigate the reconstruction of both the foreground and background. The idea presented in this part can be extended to help solve many other mathematical problems in image processing in the future.

- For future research on variational formulation with stochastic programming, our work will focus on embedding other powerful techniques such as deep network with generative capacity into a variational framework to cope with more complicated situations, e.g., cluttering and partial occlusions.
- For future work on the incorporation of variational methods and deep learning techniques, we will investigate more complicated problems such as segmenting salient objects from images with complex background or lower resolution by designing integration approach based models.

Bibliography

- [1] R. Acar and C. R. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. *Inverse problems*, 10(6):1217, 1994.
- [2] A. A. M. Al-Saffar, H. Tao, and M. A. Talab. Review of deep convolution neural network in image classification. In *2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*, pages 26–31. IEEE, 2017.
- [3] S. Alliney. Digital filters as absolute norm regularizers. *IEEE transactions on signal processing*, 40(6):1548–1562, 1992.
- [4] S. Alliney. A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. *IEEE transactions on signal processing*, 45(4):913–917, 1997.
- [5] L. Ambrosio and V. M. Tortorelli. Approximation of functional depending on jumps by elliptic functional via t-convergence. *Communications on Pure and Applied Mathematics*, 43(8):999–1036, 1990.
- [6] M. R. Amer, S. Yousefi, R. Raich, and S. Todorovic. Monocular extraction of 2.1 d sketch using constrained convex optimization. *International Journal of Computer Vision*, 112(1):23–42, 2015.
- [7] P. Amorim, T. Moraes, J. Silva, and H. Pedrini. 3d adaptive histogram equalization method for medical volumes. In *VISIGRAPP (4: VISAPP)*, pages 363–370, 2018.
- [8] G. Aubert and P. Kornprobst. *Mathematical problems in image processing: partial differential equations and the calculus of variations*, volume 147. Springer Science & Business Media, 2006.
- [9] G. Aubert and L. Vese. A variational method in image recovery. *SIAM Journal on Numerical Analysis*, 34(5):1948–1979, 1997.
- [10] J.-F. Aujol and S. H. Kang. Color image decomposition and restoration. *Journal of Visual Communication and Image Representation*, 17(4):916–928, 2006.
- [11] E. Bae, J. Shi, and X.-C. Tai. Graph cuts for curvature based image denoising. *IEEE Transactions on Image processing*, 20(5):1199–1210, 2010.
- [12] E. Bae, X.-C. Tai, and W. Zhu. Augmented lagrangian method for an euler’s elastica based segmentation model that promotes convex contours. *Inverse Problems & Imaging*, 11(1):1–23, 2017.

- [13] E. Bae, J. Yuan, X.-C. Tai, and Y. Boykov. A fast continuous max-flow approach to non-convex multi-labeling problems. In *Efficient algorithms for global optimization methods in computer vision*, pages 134–154. Springer, 2014.
- [14] M. Belkin, D. J. Hsu, and P. Mitra. Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate. In *Advances in neural information processing systems*, pages 2300–2311, 2018.
- [15] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, 2000.
- [16] J. R. Birge and F. Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.
- [17] P. Blomgren and T. F. Chan. Color tv: total variation methods for restoration of vector-valued images. *IEEE transactions on image processing*, 7(3):304–309, 1998.
- [18] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [19] X. Bresson and T. F. Chan. Fast dual minimization of the vectorial total variation norm and applications to color image processing. *Inverse problems and imaging*, 2(4):455–484, 2008.
- [20] X. Bresson, S. Esedoǧlu, P. Vandergheynst, J.-P. Thiran, and S. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and vision*, 28(2):151–167, 2007.
- [21] E.-M. Brinkmann, M. Burger, and J. S. Grah. Unified models for second-order tv-type regularisation in imaging: A new perspective based on vector operators. *Journal of Mathematical Imaging and Vision*, 61(5):571–601, 2019.
- [22] A. Brook, R. Kimmel, and N. A. Sochen. Variational restoration and edge detection for color images. *Journal of Mathematical Imaging and Vision*, 18(3):247–268, 2003.
- [23] T. Brox, M. Rousson, R. Deriche, and J. Weickert. Colour, texture, and motion in level set based segmentation and tracking. *Image and Vision Computing*, 28(3):376–390, 2010.
- [24] R. Cárdenes, R. de Luis-García, and M. Bach-Cuadra. A multidimensional segmentation evaluation for medical image data. *Computer methods and programs in biomedicine*, 96(2):108–124, 2009.
- [25] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numerische mathematik*, 66(1):1–31, 1993.
- [26] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997.
- [27] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical analysis*, 29(1):182–193, 1992.

- [28] K. H. Cha, L. Hadjiiski, R. K. Samala, H.-P. Chan, E. M. Caoili, and R. H. Cohan. Urinary bladder segmentation in ct urography using deep-learning convolutional neural network and level sets. *Medical physics*, 43(4):1882–1896, 2016.
- [29] A. Chakravarty and J. Sivaswamy. Race-net: a recurrent neural network for biomedical image segmentation. *IEEE journal of biomedical and health informatics*, 23(3):1151–1162, 2018.
- [30] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20(1-2):89–97, 2004.
- [31] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.
- [32] A. Chambolle and T. Pock. An introduction to continuous optimization for imaging. *Acta Numerica*, 25:161–319, 2016.
- [33] R. H. Chan, H. Liang, S. Wei, M. Nikolova, and X.-C. Tai. High-order total variation regularization approach for axially symmetric object tomography from a single radiograph. *Inverse Problems & Imaging*, 9(1), 2015.
- [34] T. Chan, A. Marquina, and P. Mulet. High-order total variation-based image restoration. *SIAM Journal on Scientific Computing*, 22(2):503–516, 2000.
- [35] T. F. Chan and S. Esedoglu. Aspects of total variation regularized l_1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.
- [36] T. F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM journal on applied mathematics*, 66(5):1632–1648, 2006.
- [37] T. F. Chan, B. Y. Sandberg, and L. A. Vese. Active contours without edges for vector-valued images. *Journal of Visual Communication and Image Representation*, 11(2):130–141, 2000.
- [38] T. F. Chan and J. Shen. Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, 12(4):436–449, 2001.
- [39] T. F. Chan and J. J. Shen. *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*, volume 94. Siam, 2005.
- [40] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.
- [41] T. F. Chan and L. A. Vese. A level set algorithm for minimizing the mumford-shah functional in image processing. In *Proceedings IEEE Workshop on Variational and Level Set Methods in Computer Vision*, pages 161–168. IEEE, 2001.
- [42] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of 1st International Conference on Image Processing*, volume 2, pages 168–172. IEEE, 1994.

- [43] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on image processing*, 6(2):298–311, 1997.
- [44] X. Chen, B. M. Williams, S. R. Vallabhaneni, G. Czanner, R. Williams, and Y. Zheng. Learning active contour models for medical image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11632–11640, 2019.
- [45] H. Cho, H. Lee, H. Kang, and S. Lee. Bilateral texture filtering. *ACM Transactions on Graphics (TOG)*, 33(4):1–8, 2014.
- [46] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3d unet: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.
- [47] T. Clark, A. Wong, M. A. Haider, and F. Khalvati. Fully deep convolutional neural networks for segmentation of the prostate gland in diffusion-weighted mr images. In *International Conference Image Analysis and Recognition*, pages 97–104. Springer, 2017.
- [48] D. Cremers. Image segmentation with shape priors: Explicit versus implicit representations. *Handbook of Mathematical Methods in Imaging*, 2:1909–1944, 2015.
- [49] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007.
- [50] L.-J. Deng, R. Glowinski, and X.-C. Tai. A new operator splitting method for the euler elastica model for image smoothing. *SIAM Journal on Imaging Sciences*, 12(2):1190–1230, 2019.
- [51] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.
- [52] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, pages 391–407. Springer, 2016.
- [53] J. A. Dowling, J. Fripp, S. Chandra, J. P. W. Pluim, J. Lambert, J. Parker, J. Denham, P. B. Greer, and O. Salvado. Fast automatic multi-atlas segmentation of the prostate from 3d mr images. In *International Workshop on Prostate Cancer Imaging*, pages 10–21. Springer, 2011.
- [54] J. Duan, Z. Qiu, W. Lu, G. Wang, Z. Pan, and L. Bai. An edge-weighted second order variational model for image decomposition. *Digital Signal Processing*, 49:162–181, 2016.
- [55] V. Duval, J.-F. Aujol, and L. Vese. A projected gradient algorithm for color image decomposition. *UCLA CAM report*, 2008.

- [56] S. Esedoglu and J. Shen. Digital inpainting based on the mumford–shah–euler image model. *European Journal of Applied Mathematics*, 13(4):353–370, 2002.
- [57] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. In *International conference on medical image computing and computer-assisted intervention*, pages 130–137. Springer, 1998.
- [58] Y. Gandelsman, A. Shocher, and M. Irani. Double-dip: Unsupervised image decomposition via coupled deep-image-priors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 6, page 2, 2019.
- [59] D. Geiger, H. Pao, and N. Rubin. Salient and multiple illusory surfaces. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. 98CB36231)*, pages 118–124. IEEE, 1998.
- [60] S. Geman. Statistical methods for tomographic image reconstruction. *Bull. Int. Stat. Inst.*, 4:5–21, 1987.
- [61] R. Glowinski, T.-W. Pan, and X.-C. Tai. Some facts about operator-splitting and alternating direction methods. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 19–94. Springer, 2016.
- [62] B. Goldluecke and D. Cremers. Introducing total curvature for image processing. In *2011 International Conference on Computer Vision*, pages 1267–1274. IEEE, 2011.
- [63] T. Goldstein, B. O’Donoghue, S. Setzer, and R. Baraniuk. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences*, 7(3):1588–1623, 2014.
- [64] R. Gonzalez and R. Woods. Digital image processing: Pearson prentice hall. *Upper Saddle River, NJ*, 1, 2008.
- [65] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [66] P. J. Green. Bayesian reconstructions from emission tomography data using a modified em algorithm. *IEEE transactions on medical imaging*, 9(1):84–93, 1990.
- [67] L. Gun, L. Cuihua, Z. Yingpan, and H. Feijiang. An improved speckle-reduction algorithm for sar images based on anisotropic diffusion. *Multimedia Tools and Applications*, 76(17):17615–17632, 2017.
- [68] Y. Guo, Y. Gao, and D. Shen. Deformable mr prostate segmentation via deep feature learning and sparse patch matching. *IEEE transactions on medical imaging*, 35(4):1077–1089, 2015.
- [69] Y. Han, W.-W. Wang, and X.-C. Feng. A new fast multiphase image segmentation algorithm based on nonconvex regularizer. *Pattern Recognition*, 45(1):363–372, 2012.
- [70] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

- [71] T. Hebert and R. Leahy. A generalized em algorithm for 3-d bayesian reconstruction from poisson data using gibbs priors. *IEEE transactions on medical imaging*, 8(2):194–202, 1989.
- [72] M. Hong and Z.-Q. Luo. On the linear convergence of the alternating direction method of multipliers. *Mathematical Programming*, 162(1-2):165–199, 2017.
- [73] G. Hou, J. Li, G. Wang, H. Yang, B. Huang, and Z. Pan. A novel dark channel prior guided variational framework for underwater image restoration. *Journal of Visual Communication and Image Representation*, 66:102732, 2020.
- [74] G. Hou, Z. Pan, G. Wang, H. Yang, and J. Duan. An efficient nonlocal variational method with application to underwater image restoration. *Neurocomputing*, 369:106–121, 2019.
- [75] Y. Huang, H. Yao, S. Zhao, and Y. Zhang. Towards more efficient and flexible face image deblurring using robust salient face landmark detection. *Multimedia Tools and Applications*, 76(1):123–142, 2017.
- [76] L. Hubert and P. Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.
- [77] P. Jaccard. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50, 1912.
- [78] X. Jiang, H. Yao, and S. Zhao. Edge-respecting image smoothing via extrema interpolation. In *Pacific-Rim Conference on Multimedia*, pages 190–199. Springer, 2013.
- [79] X. Jiang, H. Yao, and S. Zhao. Text image deblurring via two-tone prior. *Neurocomputing*, 242:1–14, 2017.
- [80] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.
- [81] M. Jung, X. Bresson, T. F. Chan, and L. A. Vese. Nonlocal mumford-shah regularizers for color image restoration. *IEEE transactions on image processing*, 20(6):1583–1598, 2010.
- [82] Y. M. Jung and J. J. Shen. First-order modeling and stability analysis of illusory contours. *Journal of Visual Communication and Image Representation*, 19(1):42–55, 2008.
- [83] S. H. Kang, W. Zhu, and J. Jianhong. Illusory shapes via corner fusion. *SIAM Journal on Imaging Sciences*, 7(4):1907–1936, 2014.
- [84] G. Kanizsa. *Organization in vision: Essays on Gestalt perception*. Praeger Publishers, 1979.
- [85] P. Kellman, N. Baker, G. Erlikhman, and H. Lu. Classification images reveal that deep learning networks fail to perceive illusory contours. *Journal of vision*, 17(10):569–569, 2017.

- [86] Y. Kim, B. Ham, M. N. Do, and K. Sohn. Structure-texture image decomposition using deep variational priors. *IEEE Transactions on Image Processing*, 28(6):2692–2704, 2018.
- [87] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.
- [88] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [89] R. Lai, X.-C. Tai, and T. F. Chan. A ridge and corner preserving model for surface restoration. *SIAM Journal on Scientific Computing*, 35(2):A675–A695, 2013.
- [90] T. H. N. Le, R. Gummadi, and M. Savvides. Deep recurrent level set for segmenting brain tumors. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 646–653. Springer, 2018.
- [91] T. H. N. Le, K. G. Quach, K. Luu, C. N. Duong, and M. Savvides. Reformulating level sets as deep recurrent neural network approach to semantic segmentation. *IEEE Transactions on Image Processing*, 27(5):2393–2407, 2018.
- [92] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [93] C. Li, C. Xu, C. Gui, and M. D. Fox. Level set evolution without re-initialization: a new variational formulation. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 430–436. IEEE, 2005.
- [94] F. Li, M. K. Ng, T. Y. Zeng, and C. Shen. A multiphase image segmentation method based on fuzzy region competition. *SIAM Journal on Imaging Sciences*, 3(3):277–299, 2010.
- [95] Y. Li, J. Qin, Y.-L. Hsin, S. Osher, and W. Liu. s-smooth: Sparsity and smoothness enhanced eeg brain tomography. *Frontiers in neuroscience*, 10:543, 2016.
- [96] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.
- [97] G. Litjens, R. Toth, W. van de Ven, C. Hoeks, S. Kerkstra, B. van Ginneken, G. Vincent, G. Guillard, N. Birbeck, J. Zhang, et al. Evaluation of prostate segmentation algorithms for mri: the promise12 challenge. *Medical image analysis*, 18(2):359–373, 2014.
- [98] J. Liu, Y. Sun, X. Xu, and U. S. Kamilov. Image restoration using total variation regularized deep image prior. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7715–7719. IEEE, 2019.
- [99] Q. Liu, X. Shen, and Y. Gu. Linearized admm for nonconvex nonsmooth optimization with convergence analysis. *IEEE Access*, 7:76131–76144, 2019.

- [100] Z. Liu, S. Wali, Y. Duan, H. Chang, C. Wu, and X.-C. Tai. Proximal admm for eulers elastica based image decomposition model. *Numer. Math. Theory Methods Appl.*, 12(2):370–402, 2018.
- [101] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [102] P. Loreti and R. March. Propagation of fronts in a nonlinear fourth order equation. *European Journal of Applied Mathematics*, 11(2):203–213, 2000.
- [103] S. Luan, C. Chen, B. Zhang, J. Han, and J. Liu. Gabor convolutional networks. *IEEE Transactions on Image Processing*, 27(9):4357–4366, 2018.
- [104] X.-G. Lv, Y.-Z. Song, S.-X. Wang, and J. Le. Image restoration with a high-order total variation minimization method. *Applied Mathematical Modelling*, 37(16-17):8210–8224, 2013.
- [105] M. Lysaker, A. Lundervold, and X.-C. Tai. Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *IEEE Transactions on image processing*, 12(12):1579–1590, 2003.
- [106] P. C. Mahalanobis. On the generalized distance in statistics. National Institute of Science of India, 1936.
- [107] A. Marquina and S. Osher. Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal. *SIAM Journal on Scientific Computing*, 22(2):387–405, 2000.
- [108] P. Martin, P. Réfrégier, F. Goudail, and F. Guérault. Influence of the noise model on level set active contour segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):799–803, 2004.
- [109] S. Masnou. Disocclusion: a variational approach using level lines. *IEEE Transactions on Image Processing*, 11(2):68–76, 2002.
- [110] S. Masnou and J.-M. Morel. Level lines based disocclusion. In *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No. 98CB36269)*, pages 259–263. IEEE, 1998.
- [111] A. Meyer, A. Mehrtash, M. Rak, D. Schindele, M. Schostak, C. Tempany, T. Kapur, P. Abolmaesumi, A. Fedorov, and C. Hansen. Automatic high resolution segmentation of the prostate from multi-planar mri. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 177–181. IEEE, 2018.
- [112] Y. Meyer. *Oscillating patterns in image processing and nonlinear evolution equations: the fifteenth Dean Jacqueline B. Lewis memorial lectures*, volume 22. American Mathematical Soc., 2001.
- [113] F. Milletari, N. Navab, and S.-A. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

- [114] J.-M. Morel, A.-B. Petro, and C. Sbert. Screened poisson equation for image contrast enhancement. *Image Processing On Line*, 4:16–29, 2014.
- [115] D. Mumford. Elastica and computer vision. In *Algebraic geometry and its applications*, pages 491–506. Springer, 1994.
- [116] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42(5):577–685, 1989.
- [117] M. Myllykoski, R. Glowinski, T. Karkkainen, and T. Rossi. A new augmented lagrangian approach for l^1 -mean curvature image denoising. *SIAM Journal on Imaging Sciences*, 8(1):95–125, 2015.
- [118] M. K. Ng, X. Yuan, and W. Zhang. Coupled variational image decomposition and restoration model for blurred cartoon-plus-texture images with missing pixels. *IEEE Transactions on Image Processing*, 22(6):2233–2246, 2013.
- [119] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. application to the processing of outliers. *SIAM Journal on Numerical Analysis*, 40(3):965–994, 2002.
- [120] M. Nikolova. Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares. *Multiscale Modeling & Simulation*, 4(3):960–991, 2005.
- [121] M. Nitzberg, D. Mumford, and T. Shiota. *Filtering, segmentation and depth*, volume 662. Springer, 1993.
- [122] M. Nitzberg and D. B. Mumford. *The 2.1-D sketch*. IEEE Computer Society Press, 1990.
- [123] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.
- [124] H. Pan, J. Song, W. Liu, L. Li, G. Zhou, L. Tan, and S. Chen. Using the split bregman algorithm to solve the self-repelling snake model. *arXiv preprint arXiv:2003.12693*, 2020.
- [125] K. Papafitsoros and C.-B. Schönlieb. A combined first and second order variational approach for image reconstruction. *Journal of mathematical imaging and vision*, 48(2):308–338, 2014.
- [126] N. Paragios, Y. Chen, and O. D. Faugeras. *Handbook of mathematical models in computer vision*. Springer Science & Business Media, 2006.
- [127] N. Paragios and R. Deriche. Geodesic active regions: A new framework to deal with frame partition problems in computer vision. *Journal of Visual Communication and Image Representation*, 13(1-2):249–268, 2002.
- [128] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7):629–639, 1990.

- [129] T. Poscoliero and M. Girelli. Electrophysiological modulation in an effort to complete illusory figures: configuration, illusory contour and closure effects. *Brain topography*, 31(2):202–217, 2018.
- [130] A. Prékopa. *Stochastic programming*, volume 324. Springer Science & Business Media, 2013.
- [131] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
- [132] H. K. Rafsanjani, M. H. Sedaaghi, and S. Saryazdi. An adaptive diffusion coefficient selection for image denoising. *Digital Signal Processing*, 64:71–82, 2017.
- [133] D. L. Ringach and R. Shapley. Spatial and temporal properties of illusory contours and amodal boundary completion. *Vision research*, 36(19):3037–3050, 1996.
- [134] R. T. Rockafellar and J. Sun. Solving monotone stochastic variational inequalities and complementarity problems by progressive hedging. *Mathematical Programming*, 174(1-2):453–471, 2019.
- [135] R. T. Rockafellar and R. J. Wets. Stochastic variational inequalities: single-stage to multistage. *Mathematical Programming*, 165(1):331–360, 2017.
- [136] R. T. Rockafellar and R. J.-B. Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of operations research*, 16(1):119–147, 1991.
- [137] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [138] O. Ronneberger, P. Fischer, and T.-n. Brox. Convolutional networks for biomedical image segmentation. In *Paper presented at: International Conference on Medical Image Computing and Computer-Assisted Intervention2015*.
- [139] G. Rosman, X.-C. Tai, L. Dascal, and R. Kimmel. Polyakov action minimization for efficient color image processing. In *European Conference on Computer Vision*, pages 50–61. Springer, 2010.
- [140] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.
- [141] M. Samiee, G. Thomas, and R. Fazel-Rezai. Semi-automatic prostate segmentation of mr images based on flow orientation. In *2006 IEEE International Symposium on Signal Processing and Information Technology*, pages 203–207. IEEE, 2006.
- [142] C. Samson, L. Blanc-Féraud, G. Aubert, and J. Zerubia. Multiphase evolution and variational image classification. 1999.
- [143] G. Sapiro and D. L. Ringach. Anisotropic diffusion of multivalued images with applications to color filtering. *IEEE transactions on image processing*, 5(11):1582–1586, 1996.

- [144] A. Sarti, R. Malladi, and J. A. Sethian. Subjective surfaces: A method for completing missing boundaries. *Proceedings of the National Academy of Sciences*, 97(12):6258–6263, 2000.
- [145] A. Sarti, R. Malladi, and J. A. Sethian. Subjective surfaces: A geometric model for boundary completion. *International Journal of Computer Vision*, 46(3):201–221, 2002.
- [146] A. Sawatzky, D. Tenbrinck, X. Jiang, and M. Burger. A variational framework for region-based segmentation incorporating physical noise models. *Journal of Mathematical Imaging and Vision*, 47(3):179–209, 2013.
- [147] A. Sawatzky, Q. Xu, C. O. Schirra, and M. A. Anastasio. Proximal admm for multi-channel image reconstruction in spectral x-ray ct. *IEEE transactions on medical imaging*, 33(8):1657–1668, 2014.
- [148] O. Scherzer. *Handbook of mathematical methods in imaging*. Springer Science & Business Media, 2010.
- [149] F. Shang, Y. Liu, and J. Cheng. Generalized higher-order tensor decomposition via parallel admm. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.
- [150] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2014.
- [151] J. Shen, S. H. Kang, and T. F. Chan. Euler’s elastica and curvature-based inpainting. *SIAM journal on Applied Mathematics*, 63(2):564–592, 2003.
- [152] M. Shin, M. Kim, and D.-S. Kwon. Baseline cnn structure analysis for facial expression recognition. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 724–729. IEEE, 2016.
- [153] J. Sun, H. Xu, and M. Zhang. A new interpretation of the progressive hedging algorithm for multistage stochastic minimization problems. *Journal of Industrial & Management Optimization*, 13(5):1, 2019.
- [154] M. Sussman and E. Fatemi. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on scientific computing*, 20(4):1165–1191, 1999.
- [155] M. Sussman, P. Smereka, S. Osher, et al. A level set approach for computing solutions to incompressible two-phase flow. 1994.
- [156] A. A. Taha and A. Hanbury. An efficient algorithm for calculating the exact hausdorff distance. *IEEE transactions on pattern analysis and machine intelligence*, 37(11):2153–2163, 2015.
- [157] A. A. Taha and A. Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):29, 2015.
- [158] X.-C. Tai. Fast numerical schemes related to curvature minimization: a brief and elementary review. *Actes des rencontres du CIRM*, 3(1):17–30, 2013.

- [159] X.-C. Tai and J. Duan. A simple fast algorithm for minimization of the elastica energy combining binary and level set representations. *Int. J. Numer. Anal. Model.*, 14(6):809–821, 2017.
- [160] X.-C. Tai, J. Hahn, and G. J. Chung. A fast algorithm for euler’s elastica model using augmented lagrangian method. *SIAM Journal on Imaging Sciences*, 4(1):313–344, 2011.
- [161] L. Tan, L. Li, S. An, and Z. Pan. Nonlinear diffusion based image segmentation using two fast algorithms. *Mathematical Foundations of Computing*, 2(2):149, 2019.
- [162] L. Tan, L. Li, W. Liu, J. Sun, and M. Zhang. A novel eulers elastica-based segmentation approach for noisy images using the progressive hedging algorithm. *Journal of Mathematical Imaging and Vision*, 62(1):98–119, 2020.
- [163] L. Tan, A. Liang, L. Li, W. Liu, H. Kang, and C. Chen. Automatic prostate segmentation based on fusion between deep network and variational methods. *Journal of X-ray science and technology*, 27(5):821–837, 2019.
- [164] L. Tan, W. Liu, L. Li, and Z. Pan. A fast computational approach for illusory contour reconstruction. *Multimedia Tools and Applications*, 78(8):10449–10472, 2019.
- [165] L. Tan, W. Liu, and Z. Pan. Color image restoration and inpainting via multi-channel total curvature. *Applied Mathematical Modelling*, 61:280–299, 2018.
- [166] L. Tan, Z. Pan, W. Liu, J. Duan, W. Wei, and G. Wang. Image segmentation with depth information via simplified variational level set formulation. *Journal of Mathematical Imaging and Vision*, 60(1):1–17, 2018.
- [167] L. Tan, W. B. Wei, Z. K. Pan, W. Z. Zhang, and J. M. Duan. A high-order model of tv and its augmented lagrangian algorithm. In *Applied Mechanics and Materials*, volume 568, pages 726–733. Trans Tech Publ, 2014.
- [168] Prostate MR Image Database. *The Brigham and Women’s Hospital*. (Online at: <http://prostatemrimagedatabase.com/>), 2008.
- [169] N. J. Tustison, B. B. Avants, P. A. Cook, Y. Zheng, A. Egan, P. A. Yushkevich, and J. C. Gee. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310–1320, 2010.
- [170] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.
- [171] L. A. Vese and T. F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3):271–293, 2002.
- [172] L. A. Vese and S. J. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. *Journal of scientific computing*, 19(1-3):553–572, 2003.

- [173] L. A. Vese and S. J. Osher. Color texture modeling and color image decomposition in a variational-pde approach. In *2006 Eighth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 103–110. IEEE, 2006.
- [174] C. R. Vogel and M. E. Oman. Iterative methods for total variation denoising. *SIAM Journal on Scientific Computing*, 17(1):227–238, 1996.
- [175] S. W. Wallace and W. T. Ziemba. *Applications of stochastic programming*. SIAM, 2005.
- [176] B. Wang, X. Yuan, X. Gao, X. Li, and D. Tao. A hybrid level set with semantic shape constraint for object segmentation. *IEEE transactions on cybernetics*, 49(5):1558–1569, 2018.
- [177] X. Wang and S. H. Chan. Parameter-free plug-and-play admm for image restoration. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1323–1327. IEEE, 2017.
- [178] X. Wang, B. Zhang, C. Li, R. Ji, J. Han, X. Cao, and J. Liu. Modulated convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 840–848, 2018.
- [179] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.
- [180] C. Wu and X.-C. Tai. Augmented lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models. *SIAM Journal on Imaging Sciences*, 3(3):300–339, 2010.
- [181] C. Yan, H. Xie, J. Chen, Z. Zha, X. Hao, Y. Zhang, and Q. Dai. A fast uyghur text detector for complex background images. *IEEE Transactions on Multimedia*, 20(12):3389–3398, 2018.
- [182] C. Yan, H. Xie, D. Yang, J. Yin, Y. Zhang, and Q. Dai. Supervised hash coding with deep neural network for environment perception of intelligent vehicles. *IEEE transactions on intelligent transportation systems*, 19(1):284–295, 2017.
- [183] C. Yan, Y. Zhang, J. Xu, F. Dai, L. Li, Q. Dai, and F. Wu. A highly parallel framework for hevcc coding unit partitioning tree decision on many-core processors. *IEEE Signal Processing Letters*, 21(5):573–576, 2014.
- [184] C. Yan, Y. Zhang, J. Xu, F. Dai, J. Zhang, Q. Dai, and F. Wu. Efficient parallel framework for hevcc motion estimation on many-core processors. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(12):2077–2089, 2014.
- [185] S. Yan, X.-c. Tai, J. Liu, and H.-y. Huang. Convexity shape prior for level set based image segmentation method. *arXiv preprint arXiv:1805.08676*, 2018.
- [186] Y. Yan, F. Nie, W. Li, C. Gao, Y. Yang, and D. Xu. Image classification by cross-media active learning with privileged information. *IEEE Transactions on Multimedia*, 18(12):2494–2502, 2016.

- [187] F. Yang, A. M. Bruckstein, and L. D. Cohen. Pointflow: A model for automatically tracing object boundaries and inferring illusory contours. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 485–498. Springer, 2017.
- [188] F. Yang, K. Chen, B. Yu, and D. Fang. A relaxed fixed point method for a mean curvature-based denoising model. *Optimization Methods and Software*, 29(2):274–285, 2014.
- [189] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM Journal on Imaging Sciences*, 2(2):569–592, 2009.
- [190] M. Yashtini and S. H. Kang. Alternating direction method of multiplier for euler’s elastica-based denoising. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 690–701. Springer, 2015.
- [191] M. Yashtini and S. H. Kang. A fast relaxed normal two split method and an effective weighted tv approach for euler’s elastica image inpainting. *SIAM Journal on Imaging Sciences*, 9(4):1552–1581, 2016.
- [192] A. M. Yip and W. Zhu. A fast modified newton’s method for curvature based denoising of 1d signals. *Inverse Probl. Imaging*, 7(3):1075–1097, 2013.
- [193] Y.-L. You and M. Kaveh. Fourth-order partial differential equations for noise removal. *IEEE Transactions on Image Processing*, 9(10):1723–1730, 2000.
- [194] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.
- [195] L. Yu, X. Yang, H. Chen, J. Qin, and P. A. Heng. Volumetric convnets with mixed residual connections for automated prostate segmentation from 3d mr images. In *Thirty-first AAAI conference on artificial intelligence*, 2017.
- [196] Y. Yu, Z. Pan, W. Wei, and J. Jiang. Edge preserving of some variational models for vectorial image denoising. *J. Gr. Images*, 16(12):2223–2230, 2011.
- [197] J. Yuan, E. Bae, and X.-C. Tai. A study on continuous max-flow and min-cut approaches. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2217–2224. IEEE, 2010.
- [198] A. Zaim and J. Jankun. An energy-based segmentation of prostate from ultrasound images using dot-pattern select cells. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP’07*, volume 1, pages I–297. IEEE, 2007.
- [199] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [200] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2528–2535. IEEE, 2010.

- [201] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- [202] H. Zhang, L. Tang, and C. He. A variational level set model for multiscale image segmentation. *Information Sciences*, 493:152–175, 2019.
- [203] K. Zhang, W. Zuo, S. Gu, and L. Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017.
- [204] H. Zhao. A fast sweeping method for eikonal equations. *Mathematics of computation*, 74(250):603–627, 2005.
- [205] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of computational physics*, 127(1):179–195, 1996.
- [206] Q. Zhong, Y. Li, Y. Yang, and Y. Duan. Minimizing discrete total curvature for image processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9474–9482, 2020.
- [207] Z. Zhou, Z. Guo, D. Zhang, and B. Wu. A nonlinear diffusion equation-based model for ultrasound speckle noise removal. *Journal of Nonlinear Science*, 28(2):443–470, 2018.
- [208] W. Zhu and T. Chan. A variational model for capturing illusory contours using curvature. *Journal of Mathematical Imaging and Vision*, 27(1):29–40, 2007.
- [209] W. Zhu and T. Chan. Image denoising using mean curvature of image surface. *SIAM Journal on Imaging Sciences*, 5(1):1–32, 2012.
- [210] W. Zhu, T. Chan, and S. Esedoğlu. Segmentation with depth: A level set approach. *SIAM journal on scientific computing*, 28(5):1957–1973, 2006.
- [211] W. Zhu, X.-C. Tai, and T. Chan. Augmented lagrangian method for a mean curvature based image denoising model. *Inverse Probl. Imaging*, 7(4):1409–1432, 2013.
- [212] W. Zhu, X.-C. Tai, and T. Chan. Image segmentation using eulers elastica as the regularization. *Journal of scientific computing*, 57(2):414–438, 2013.
- [213] Y. Zhu, S. Williams, and R. Zwigelaar. A hybrid asm approach for sparse volumetric data segmentation. *Pattern recognition and image analysis*, 17(2):252–258, 2007.

Every reasonable effort has been made to acknowledgement the owners of copyright material. I would be pleased to hear from any copyright owner who has been omitted or incorrectly acknowledged.