University of Nottingham
UK | CHINA | MALAYSIA

Incorporating Fuzzy-based Methods to Deep Learning Models for Semantic Segmentation

This thesis is submitted for the degree of
Doctor of Philosophy

Qiao Lin

20200056

Supervised by
Prof. Jonathan M Garibaldi
Dr Xin Chen
Dr Chao Chen

School of Computer Science

University of Nottingham

Submitted November 2023

Abstract

This thesis focuses on improving the workflow of semantic segmentation through a combination of reducing model complexity, improving segmentation accuracy, and making semantic segmentation results more reliable and robust. Semantic segmentation refers to pixel-level classification, the objective of which is to classify each pixel of the input image into different categories. The process typically consists of three steps: model construction, training, and application. Thus, in this thesis, fuzzy-based techniques are utilized in the aforementioned three steps to improve semantic segmentation workflow .

The widely-used semantic segmentation models normally extract and aggregate spatial information and channel-wise features simultaneously. In order to achieve promising segmentation performance, it is required to involve numerous learnable parameters, which increase the model's complexity. Thus, decoupling the information fusion tasks is an important approach in the exploration of semantic segmentation models. Fuzzy integrals are effective for fusing information, and some special fuzzy integral operators (OWA) are free of parameters and easy to implement in deep-learning models. Therefore, a novel fuzzy integral module that includes an additional convolutional layer for feature map dimensionality reduction and an OWA layer for information fusion across feature channels is designed. The proposed fuzzy integral module can be flexibly integrated into existing semantic segmentation models, and then help reduce parameters and save memory.

Following the exploration of semantic segmentation models, the collected data is used to train the model. Note that the precise delineation of object boundaries is a key aspect of semantic segmentation. In order to make the segmentation model pay more attention to the boundary, a special boundary-wise loss function is desirable in the segmentation model training phase. Fuzzy rough sets are normally utilized to measure the relationship between two sets. Thus, in this thesis, to improve the boundary accuracy, fuzzy rough sets are leveraged to calculate a boundary-wise loss, which is the difference between the boundary sets

of the predicted image and the ground truth image.

After completing the training process with the proposed novel loss, the next step for semantic segmentation is to apply the pre-trained segmentation model to segment new images. One challenge is that there are no ground truth images to quantify the segmentation quality in the real-world application of semantic segmentation models. Therefore, it is crucial to design a quality quantification algorithm to infer image-level segmentation performance and improve the credibility of semantic segmentation models. In this thesis, a novel quality quantification algorithm based on fuzzy uncertainty is proposed as part of the model inference process without accessing ground truth images.

Moreover, to further explore the practical application of the proposed quality quantification algorithm in clinical settings, this thesis goes beyond public datasets and delves into a real-world case study involving cardiac MRI segmentation. Additionally, as clinicians also provide the level of uncertainty to measure their confidence when annotating to generate ground truth images (human-based uncertainty), the correlation between human-based uncertainty and AI-based uncertainty (calculated by the proposed quality quantification algorithm) is deeply investigated.

Comprehensive experiments are conducted in this thesis to demonstrate that the integration of fuzzy-based technologies can enhance the efficiency, accuracy, and reliability of semantic segmentation models compared to those without such methods.

Declaration

I hereby declare that the work in this thesis is original and has not been submitted for any other degree or university. This thesis is my own work and based on the research at LUCID and IMA labs, in the School of Computer Science, the University of Nottingham, United Kingdom.

Qiao Lin
20200056
November 2023

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Prof. Jonathan M. Garibaldi, Dr. Xin Chen, and Dr. Chao Chen. I feel fortunate to work with them. Their support and patience gave me a fulfilling and enjoyable Ph.D. life.

I would like to extend my thanks to the members of our research groups LUCID and IMA: Prof. Christian Wagner, Dr. Direnc Pekaslan, Ruizhe Li, Han Meng, and Te Zhang.

Finally, I am deeply grateful to my family and friends for their continuous support and encouragement. When I feel depressed, my parents and sisters always help me release negative emotions and cheer me up.

Contents

Abstract i

Declaration iii

Acknowledgements iv

1 Introduction 1

1.1 Background and Motivation 1

1.2 Aims and Objectives 5

1.3 Thesis Structure 7

1.4 Publications Arising from the Thesis 8

2 Background 10

2.1 Introduction 10

2.2 Semantic Segmentation 11

2.2.1 Semantic Segmentation Models 13

2.2.2 Semantic Segmentation Loss Functions 15

2.2.3 Quality Quantification for Semantic Segmentation 17

2.3 Fuzzy Sets 19

2.3.1 Type-1 Fuzzy Sets 20

2.3.2 Type-2 Fuzzy Sets 21

2.3.3 Fuzzy Set Operations 24

2.4 Fuzzy Rough Sets 26

2.4.1 Rough Sets 26

2.4.2 Fuzzy Rough Sets 27

2.5 Fuzzy Integral 28

2.5.1 OWA Operators 29

2.6 Summary 32

3 A Novel Fuzzy Integral Module in Semantic Segmentation Models 34

3.1 Background and Motivation 35

3.2 Fuzzy Integral Module 38

3.3 Evaluation and Results 40

3.3.1 Implementation Details 41

3.3.2 Comparison with Other Fuzzy-integral-based Semantic Segmentation Model 42

3.3.3 Application of Fuzzy Integral Module in UNet 46

3.4 Discussion 50

3.5 Summary 54

4 Boundary-wise Loss for Semantic Segmentation Based on Fuzzy Rough Sets 55

4.1 Background and Motivation 56

4.2 A New Boundary-wise Loss Function for Semantic Segmentation 58

4.2.1 Lower Approximation of Fuzzy Rough Sets 58

4.2.2 Fuzzy Rough Sets Loss 61

4.2.3 Distance Transform Algorithm 64

4.2.4 Computational Details of Fuzzy Rough Sets Loss 66

4.3 Evaluation and Results 68

4.3.1 Datasets and Metrics 69

4.3.2 Experimental Design 71

4.3.3 Results for Nuclei Dataset 73

4.3.4 Results for Cell Dataset 78

4.4 Discussion 81

4.5 Summary 87

5 A Novel Quality Quantification Algorithm for Semantic Segmentation Based on Fuzzy Uncertainty 89

5.1 Background and Motivation 90
5.2 A Novel Quality Quantification Method 92
5.2.1 DCNN-based Image Segmentation Model 93
5.2.2 Test-Time Augmentation and Monte Carlo Dropout . . . 94
5.2.3 Grouped Distance Map Generation 96
5.2.4 Fuzzy Sets Generation 100
5.2.5 Image-level Uncertainty Estimation 104
5.3 Evaluation and Results 104
5.3.1 Parameter Settings 108
5.3.2 Comparison of Uncertainty-based Quality Quantification Methods 111
5.3.3 Application of Uncertainty for Quality Quantification . . 116
5.4 Discussion 118
5.5 Summary 122
6 Real-world Medical Case Study **124**
6.1 Background and Motivation 125
6.2 Implementations Details 127
6.2.1 Materials 127
6.2.2 Segmentation Model 128
6.2.3 AI-derived Uncertainty Generation 129
6.3 Uncertainty-based Quality Quantification 131
6.4 Comparison of AI-based Uncertainty and Human-based Uncertainty 134
6.5 Qualitative Analysis 141
6.5.1 Clinical Application of AI-derived Uncertainty 141
6.5.2 Generation of Human-based Uncertainty 142
6.6 Discussion 145
6.7 Summary 146
7 Conclusions **148**

7.1 Thesis Summary 148
7.2 Contributions and Publications 150
7.3 Limitations 154
7.4 Future Work 156
Bibliography **159**
Appendices **176**
A Good Segmentation Case **178**
A.1 Whole 3D Image 178
A.2 Each Slice 179
B Middle Segmentation Case **184**
B.1 Whole 3D Image 184
B.2 Each Slice 185
C Poor Segmentation Case **188**
C.1 Whole 3D Image 188
C.2 Each Slice 189

List of Tables

2.1 Examples for s-norm operators 26
2.2 Examples for t-norm operators 26

3.1 Experimental results for baseline model, Price's fuzzy models and the proposed fuzzy models. Mean $\pm$ standard deviation values are reported for Dice measure. The number of model parameters are also listed. $\sharp$ means the proposed model and Price's model are significantly different measured by wilcoxon sign rank test with P value < 0.05; $*$ means the proposed model and the baseline model are significantly different with P value < 0.05 45
3.2 Experimental results for the widely-used segmentation model UNet with the proposed fuzzy integral module. 48

4.1 Experimental results for boundary-wise losses, pixel-wise loss and region-wise loss in UNet, FCN and SegNet on the nuclei dataset. Mean $\pm$ standard deviation values are reported for all the evaluation measures. $*$ represents FRLoss and CELoss are significantly different with $p < 0.05$; $\sharp$ represents FRLoss and DLoss are significantly different with $p < 0.05$ 74
4.2 Convergence time of the UNet, FCN and SegNet models with different boundary-wise losses 76

4.3 Experimental results for boundary-wise losses, pixel-wise loss and region-wise loss in UNet, FCN and SegNet on the cell dataset. Mean $\pm$ standard deviation values are reported for all the evaluation measures. $*$ represents FRLoss and CELoss are significantly different with $p < 0.05$; $\sharp$ represents FRLoss and DLoss are significantly different with $p < 0.05$ 78

5.1 Experimental results for different uncertainty-based quality quantification methods based on five-fold cross-validation. The mean and standard deviation values of the five results are calculated to represent the Pearson correlation between measured uncertainty and the Dice coefficient in terms of Mean $\pm$ standard deviation. 112

5.2 The average computational time of FQC and FQCfuzzy for one test image on the five given datasets 119

5.3 The influence of poor-quality segmentation results on IoU, Dpw and FQC methods 121

6.1 The true segmentation quality of cardiac MRI dataset and the Pearson correlation coefficient (PE) between DC and AI-based Uncertainty 134

6.2 The statistic values for each box of slice-level DC; $*$ means box-1 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05 135

6.3 The statistic values for each box of slice-level AI-based uncertainty; $\sharp$ means box-0 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05; $*$ means box-1 and box-2 are significantly different with P value <0.05 136

6.4 The statistic values for each box of slice-class-level AI-based uncertainty; ♯ means box-0 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05; $*$ means box-1 and box-2 are significantly different with P value <0.05; ‡ means box-0 and box-1 are significantly different with P value <0.05 140

A1 The results for the whole 20CA015_N133_SAX.nii.gz Image . . 178

B1 The results for the whole 20CA015_N213_SAX.nii.gz Image . . 184

C1 The results for the whole 20CA015_N064_SAX.nii.gz Image . . 188

List of Figures

2.1 Link each topic to the aims and objectives 11

2.2 The Process of Semantic Segmentation 12

2.3 The structure of UNet model [1] 14

2.4 The structure of SegNet model [2] 14

2.5 Example of type-1 fuzzy sets [3] 21

2.6 Example of type-2 fuzzy sets [4] 23

2.7 Example of interval fuzzy sets [5]. The solid line is the upper boundary of FOU and refers to the upper membership function (UMF). The dotted line is the lower boundary of FOU and represents the lower membership function (LMF). The shaded area is the FOU. 24

3.1 The structure of the fuzzy integral module 38

3.2 (a) The architecture of the CNN model: grey blocks and white arrows mean the down-sampling process, blue blocks and arrows represent up-sampling process, red arrows stand for the concrete location of fuzzy module. When there is no fuzzy module, the CNN model is called baseline model; (b) This part is the details of fuzzy module including dimensionality reduction layers and fuzzy fusion layers. N_r means the number of feature maps after applying the dimensionality reduction operator. N_f represents the number of fuzzy integral operators. Each fuzzy fusion operator generates a new feature map. Black arrows mean the maxpooling operator for the baseline model and the number of output feature maps is 64. Black dashed arrows mean the maxpooling operator for fuzzy models and the number of output feature maps is N_f. 43

3.3 Segmentation results: (a) original image; (b) benchmark image; (c) predicted image segmentation of baseline model; (d) predicted image segmentation of 'Down_fuzzy' model based on Price's method [6]; (e) predicted image segmentation of 'Down_-fuzzy' model based on the proposed method; (f) predicted image segmentation of 'Up_fuzzy' model based on Price's method [6]; (g) predicted image segmentation of 'Up_fuzzy' model based on the proposed method; (h) predicted image segmentation of 'All_-fuzzy' model based on Price's method [6]; (i) predicted image segmentation of 'All_fuzzy' model based on the proposed method. 47

3.4 The pipeline of UNet 48

3.5 The weights updating process of one filter for the proposed model and the Price's model [6] (a) means that all of the chosen weights in the proposed new model are updated regularly, (b) shows that the chosen weights in the Price's model [6] are not successively updated. 51

3.6 The visualization of the fusion process of two elements 53

4.1 The left part refers to the predicted image and the right part is the ground truth image. Black circle is the predicted image boundary pixel set; orange circle is the ground truth image boundary pixel set and belongs to the class D_1; the rest parts of the ground truth image pixels belong to the class D_2; grey part is the area of segmentation object. 61

4.2 The blue line is the object boundary of the ground truth image; the red line is the object boundary of the predicted segmentation image; the grey region is the overlap of the ground truth image and the predicted segmentation image; the yellow region is disjoint parts of the ground truth image and the predicted segmentation image. 66

4.3 Segmentation performance with different σ, when $\sigma = 1$, the semantic segmentation model achieves the best performance. . . 73

4.4 Boundary distance metrics ASD and 95th-percentile of HD for DLoss, FRSLoss and CELoss on Nuclei dataset. When the values of ASD and 95th-percentile of HD are lower, the predicted image and the ground truth image would have closer boundary distance. 77

4.5 Boundary distance metrics ASD and 95th-percentile of HD for DLoss, FRSLoss, and CELoss on Cell dataset. When the values of ASD and 95th-percentile of HD are lower, the predicted image and the ground truth image would have closer boundary distance. 80
4.6 Segmentation results for two datasets and three models: (a) original image; (b) benchmark image; (c) predicted segmentation image of FRSLoss; (d) predicted segmentation image of HDLoss; (e) predicted segmentation image of DFRSLoss; (f) predicted segmentation image of DHDLoss;(g) predicted segmentation image of CELoss; (h) predicted segmentation image of DLoss. The red circle and arrow represent the boundary difference between FRSLoss, CELoss, and DLoss. 81
4.7 Loss variation curves for UNet, FCN and SegNet 82
4.8 Loss variation curves for UNet, FCN and SegNet 83
4.9 Loss variation curves for UNet, FCN and SegNet 84
4.10 Loss variation curves for UNet, FCN and SegNet 85

5.1 The flow chart for the fuzzy-uncertainty-based quality quantification algorithm. It consists of four steps: (1) uncertainty estimation using MCdropout and test-time augmentation; (2) distance map generation; (3) fuzzy-set generation (4) Image level uncertainty estimation. 93
5.2 The pipeline to calculate the grouped distance map from predicted images obtained with test-time augmentation and MCdropout. 97

5.3 Three membership functions, (a) maps the small pixel values to high membership values, (b) maps the medium pixel values to high membership values, (c) maps the large pixel values to high membership values. 100
5.4 The pipeline to obtain the fuzzy sets 101
5.5 The curve of PCC with different K values when N is equal to (a) 12, (b) 24, and (c) 48 respectively 109
5.6 The curve of PCC with different N values when K is equal to (a) 5, (b) 16, and (c) 40 respectively 110
5.7 Scatter plots for the relationship between the uncertainty-based quality quantification methods and the segmentation quality measurement (the Dice coefficient). Each points refers to one test image for the given five datasets. The sky-blue dots line refers to the threshold (Dice $= 0.8$) of truly good or poor segmentation images. The red dots line means the threshold for the six uncertainty-based quality quantification methods to classify the good or poor segmentation images. Note that as FQC, VC and Ulabelled have a negative relationship with the Dice coefficient, the left part of the red dots line is the predicted good segmentation images and the right part of the red dots line is the predicted poor segmentation images. Whereas IoU, Dpw and CNNurp have a positive relationship with the Dice coefficient, thus the right part of the red dots line is the predicted good segmentation images and the left part of the red dots line is the predicted poor segmentation images. 114

5.8 The process of searching the optimal threshold. Yellow points indicate the truly good quality segmentation images and green points indicate the truly poor quality segmentation images. The red line indicates the threshold and can be considered as a binary classifier. The optimal threshold represents the classifier has the best performance. 117
5.9 The ROC curves for FQCfuzzy, FQC, VC, Ulabelled, IoU, Dpw and CNNurp. The area under the ROC curve namely AUC is a a criterion to measure the classification capability of the classifier. 118

6.1 The Pipeline of the AI-based uncertainty algorithm 129
6.2 Scatter plot for the relationship between the DC and AI-based uncertainty. PE represents the Pearson correlation between AI-based uncertainty and DC. 133
6.3 The box plot of human-based uncertainty and the true slice-level segmentation quality (measured by slice-level DC) 135
6.4 The box plot for AI-based uncertainty and Human-based uncertainty on the slice level. 137
6.5 The box plot of AI-based Uncertainty and Human-based Uncertainty for each class. 138
6.6 (a) is the box plot for the relationship between human-based uncertainty and slice-level structure' size. (b) is the box plot for the relationship between AI-based uncertainty and structure' size at the slice level. 139
6.7 Two cases with low AI-based uncertainty (0.085 and 0.095) and high human-based uncertainty (level 0). The tissue is the Aor. . . 143
6.8 Two cases with low AI-based uncertainty (0.085 and 0.080) and high human-based uncertainty (level 1). The tissue is the Aor. . . 143

6.9 Two cases with low AI-based uncertainty (0.874 and 0.821) and high human-based uncertainty (level 2). The tissues are LV-En (red colour), LV-Ep (green colour), and Scar (Blue colour). . . . 144

A1 The results for slice-1 of 20CA015_N133_SAX.nii.gz 179
A2 The results for slice-2 of 20CA015_N133_SAX.nii.gz 179
A3 The results for slice-3 of 20CA015_N133_SAX.nii.gz 180
A4 The results for slice-4 of 20CA015_N133_SAX.nii.gz 180
A5 The results for slice-5 of 20CA015_N133_SAX.nii.gz 181
A6 The results for slice-6 of 20CA015_N133_SAX.nii.gz 181
A7 The results for slice-7 of 20CA015_N133_SAX.nii.gz 182
A8 The results for slice-8 of 20CA015_N133_SAX.nii.gz 182
A9 The results for slice-9 of 20CA015_N133_SAX.nii.gz 182
A10 The results for slice-10 of 20CA015_N133_SAX.nii.gz 183
A11 The results for slice-11 of 20CA015_N133_SAX.nii.gz 183

B1 The results for slice-1 of 20CA015_N213_SAX.nii.gz 185
B2 The results for slice-2 of 20CA015_N213_SAX.nii.gz 185
B3 The results for slice-3 of 20CA015_N213_SAX.nii.gz 185
B4 The results for slice-4 of 20CA015_N213_SAX.nii.gz 186
B5 The results for slice-5 of 20CA015_N213_SAX.nii.gz 186
B6 The results for slice-6 of 20CA015_N213_SAX.nii.gz 186
B7 The results for slice-7 of 20CA015_N213_SAX.nii.gz 187
B8 The results for slice-8 of 20CA015_N213_SAX.nii.gz 187
B9 The results for slice-9 of 20CA015_N213_SAX.nii.gz 187

C1 The results for slice-1 of 20CA015_N064_SAX.nii.gz 189
C2 The results for slice-2 of 20CA015_N064_SAX.nii.gz 189
C3 The results for slice-3 of 20CA015_N064_SAX.nii.gz 190
C4 The results for slice-4 of 20CA015_N064_SAX.nii.gz 190

C5 The results for slice-5 of 20CA015_N064_SAX.nii.gz 190
C6 The results for slice-6 of 20CA015_N064_SAX.nii.gz 191
C7 The results for slice-7 of 20CA015_N064_SAX.nii.gz 191
C8 The results for slice-8 of 20CA015_N064_SAX.nii.gz 191
C9 The results for slice-9 of 20CA015_N064_SAX.nii.gz 192
C10 The results for slice-10 of 20CA015_N064_SAX.nii.gz 192
C11 The results for slice-11 of 20CA015_N064_SAX.nii.gz 192
C12 The results for slice-11 of 20CA015_N064_SAX.nii.gz 193

Chapter 1

Introduction

1.1 Background and Motivation

Image segmentation is the process of dividing a digital image into various image objects, the goal of which is to reduce the complexity of the image and improve the efficiency of image analysis. As the digital image is represented in the form of pixels, image segmentation is equivalent to grouping pixels. The classical segmentation methods consist of thresholding [7], region growing [8], edge detection [9], and machine learning methods using handcrafted features [10]. With the development of hardware equipment and the increasing number of images, end-to-end convolutional neural networks (CNNs)–based semantic image segmentation techniques have become state-of-the-art methods. Semantic segmentation essentially refers to pixel-level classification, the objective of which is to classify each pixel of the input image into different categories and then segment the given image into various useful and meaningful regions based on the pixel classification results. In some areas (e.g., skin lesion segmentation[11], lung tumor segmentation[12]), the performance of semantic segmentation can be on par with that of human experts.

Fundamentally, the semantic segmentation process consists of three key steps: 1) construct a semantic segmentation model using deep convolutional neural networks; 2) leverage raw images and related ground truth masks to train the segmentation model and learn corresponding weights; 3) use the pre-trained segmentation model to segment new images. In order to improve the segmentation quality, researchers have proposed numerous models e.g. FCN [13], UNet [1], SegNet [2], and DeepLab series [14, 15, 16, 17]. These semantic segmentation models have a general framework, including two parts: (1) the encoding stage is utilized to extract spatial and channel information; (2) the decoding stage is applied to resize the output prediction to the same size as the corresponding ground truth mask. In these models, the spatial information and channel information of one given image are extracted simultaneously. As we know, when a model deals with multiple tasks at the same time, its performance for each individual task is potentially degraded. Thus, Hu et al. [18] designed a module named squeeze-and-excitation (SE) to handle channel information specifically, which won first place in several semantic segmentation competitions. However, one limitation of SE is that the process for the combination of channel information is a black box. Whether an interpretable fusion method can be used to aggregate channel information is still an open question. To investigate this, one of the state-of-art information fusion technologies called fuzzy integrals is introduced in Chapter 3.

After the semantic segmentation model is determined, the next step is to learn the model parameters using the collected training data. First, the raw images are sent into the pre-defined model to obtain the predicted segmentation images. Then the difference between the predicted images and the ground truth masks is calculated based on the selected loss function to update the model parameters until the model converges. Hence, the loss function plays a significant role during the model training process. Many past and current studies pay attention

to region-wise or pixel-wise losses [19, 20, 21, 22, 23]. Pixel-wise losses concentrate on local details, while region-wise losses focus on global information. However, in some specific application scenarios e.g. tumor segmentation, it is more important to precisely delineate objects' boundaries than to segment accurate pixels or regions. It indicates that a balance between local and global is necessary. Thus, a boundary-wise loss measuring the difference between the predicted image's and the ground truth mask's boundaries is proposed. To simplify the computational process, the corresponding boundaries are regarded as two sets. Considering that fuzzy rough sets with the fuzzy equivalence relation are a useful and widely-used approach to measuring the difference between two sets [24], the lower approximation of fuzzy rough sets is introduced to calculate the boundary-wise loss. More investigations on boundary-wise losses are discussed in Chapter 4.

When the training process with the chosen loss function is finished, the pre-trained segmentation model is used to segment new images. It has been demonstrated that pre-trained semantic segmentation models have achieved outstanding performance in multiple public datasets [25]. However, the applications of semantic segmentation models in the real world are still limited due to the fact that no reliable indication of the segmentation quality can be provided. Current semantic segmentation models have no ability to indicate the success/failure or the level of trustworthiness of the segmentation result. Instead, these models only provide a segmentation result without segmentation quality information, which limits the widespread application of image segmentation models especially in clinical settings. Note that the pixel-wise confidence scores provided by the segmentation models are different from the uncertainty or trustworthiness of the segmentation results. Therefore, it would be of great importance to design a quality quantification algorithm for the image segmentation models. The quality quantification algorithm should be capable of indicating whether

the segmentation result has poor or good quality without knowing the ground truth masks. As a high uncertainty value generally indicates an incorrect prediction, the segmentation quality has a negative relationship with the segmentation uncertainty [26]. It is a natural idea to leverage uncertainty to infer segmentation quality. Fuzzy sets, proposed by Zadeh in 1965 [27], can efficiently handle ambiguity and vagueness in many fields [28, 29, 30]. Thus, whether the fuzzy sets can be applied to quantify the segmentation uncertainty and then indicate the segmentation quality is still an open question. Chapter 5 provides a detailed investigation and discussion.

Throughout the thesis, different fuzzy methods have been adopted to deal with semantic segmentation issues: 1) fuzzy integrals are used to aggregate channel-wise information in semantic segmentation models thereby reducing the complexity of semantic segmentation models; 2) fuzzy rough sets are applied to calculate the boundary-wise loss, therefore to improve the boundary accuracy of semantic segmentation; 3) fuzzy uncertainty is utilized to design quality quantification algorithm and then infers the semantic segmentation performance. Furthermore, in order to explore the practical application of the proposed quality quantification method in clinical settings, Chapter 6 goes beyond public datasets and delves into a real-world case study involving cardiac MRI segmentation. Quantitative analysis and qualitative analysis are conducted to investigate how to use the proposed quality quantification method in the clinical setting and what the difference between AI-based uncertainty (calculated using the quality quantification algorithm) and human-based uncertainty (clinicians' confidence in annotating to generate ground truth images) is. The quantitative analysis is to obtain some experimental results based on the dataset, while the qualitative analysis is designed for clinicians to obtain feedback and comments regarding the application of uncertainty in real-world settings and how they annotate the uncertainty.

1.2 Aims and Objectives

The aim of this thesis is to improve the workflow of semantic segmentation through a combination of reducing model complexity, improving segmentation accuracy, and making semantic segmentation results more reliable and robust. The corresponding objectives to obtain the aim are described as follows in detail.

- *Create a modeling framework in which the number of parameters is suitably low*

  The widely-used semantic segmentation models normally extract and aggregate spatial information and channel-wise features simultaneously. In order to achieve promising segmentation performance, it is required to involve numerous learnable parameters, which increases the model's complexity. Thus, one objective of this thesis is to create a modeling framework in which the number of parameters is suitably low. Note that fuzzy integrals are effective for fusing information, and some special fuzzy integral operators are free of parameters. Applying the fuzzy integral to fuse the channel information is a logical move that can decouple the information fusion tasks in semantic segmentation models while also simplifying the models.

- *Improve overall segmentation accuracy with particular emphasis on boundaries*

  One of the key points of semantic segmentation is to precisely delineate objects' boundaries. To make the segmentation model pay more attention to the boundary, a special boundary-wise loss function should be implemented during the segmentation model training phase. Thus, one objective of this research is to improve the boundary accuracy of semantic segmentation by designing a novel boundary-wise loss. Note that fuzzy

rough sets are normally utilized to measure the relationship between two sets using fuzzy equivalence relation. Moreover, the boundary-wise loss function is to calculate the difference between the boundary sets of the predicted image and the ground truth image. Therefore, using fuzzy rough sets to design the novel boundary-wise loss is a possible direction.

- *Enhance interpretability of semantic segmentation results*

  Without interpretability, the semantic segmentation results are inconvincible and not accepted by users, especially clinicians, which limits the application of semantic segmentation. Hence, in order to make semantic segmentation results more reliable and robust, one of the objectives is to enhance the interpretability of semantic segmentation results by designing a novel quality quantification algorithm. The proposed quality quantification algorithm can help interpret the semantic segmentation results by segmentation uncertainty. Uncertainty has a negative relationship with segmentation quality, and fuzzy sets are an efficient and useful technique to handle and quantify uncertainty. It is a promising idea to leverage fuzzy sets to calculate the segmentation uncertainty and, therefore to indicate the quality of semantic segmentation results.

- *Evaluating the framework through real-world experimental studies*

  After completing the algorithms or models design in the lab, the next step is to utilize the proposed algorithms and models to handle practical issues. Hence, another objective is to evaluate the framework through real-world experimental studies by conducting quantitative analysis and qualitative analysis with clinicians.

1.3 Thesis Structure

This chapter provides an overview of this thesis. Firstly, background information regarding research gaps and the corresponding motivations is given. Then aims and objectives of this work are summarized. The following section includes a list of publications that have resulted from this thesis.

The structure of the remainder of this thesis is as follows:

- Chapter 2 outlines an overview of some background information and a literature survey for this thesis, including semantic segmentation and fuzzy techniques.
- Chapter 3 proposes a new fuzzy integral module that can be flexibly inserted into semantic segmentation models. This fuzzy integral module consists of a dimensionality reduction operator and ordered weight averaging (OWA) fusion operators, which are capable of reducing model complexity.
- Chapter 4 presents a novel boundary-wise semantic segmentation loss based on the lower approximation of fuzzy rough sets. This new loss pays more attention to the boundaries in comparison to other segmentation losses.
- Chapter 5 introduces a novel fuzzy-uncertainty-based quality quantification algorithm for semantic segmentation. This algorithm offers enhanced capabilities in assessing segmentation quality and classifying the good/poor segmentation images.
- Chapter 6 conducts quantitative analysis and qualitative analysis simultaneously on a real-world cardiac MRI dataset to investigate how to use the

proposed quality quantification method in the clinical setting and what the difference between AI-based uncertainty and human-based uncertainty is.

- Chapter 7 concludes the work in this thesis while also discussing the limitations of the current work and making suggestions for further research.

1.4 Publications Arising from the Thesis

Three conference papers and three journal papers have been published or submitted after completing the research in this thesis. These publications are as follows:

[1] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "Quality Quantification in Deep Convolutional Neural Networks for Skin Lesion Segmentation using Fuzzy Uncertainty Measurement, " in Proceedings IEEE International Conference on Fuzzy Systems, 2022, pp. 1-8.

[2] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "FuzzyDCNN: Incorporating Fuzzy Integral Layers to Deep Convolutional Neural Networks for Image Segmentation" in Proceedings IEEE International Conference on Fuzzy Systems, 2021, pp. 1-7.

[3] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi,"Quality control algorithm for medical image segmentation based on fuzzy uncertainty", IEEE Transactions on Fuzzy Systems, vol. 31, no. 8, pp. 2532-2544, 2023.

[4] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "Fuzzy Uncertainty-based Out-of-Distribution Detection Algorithm for Semantic Segmentation" in Proceedings IEEE International Conference on Fuzzy Systems, 2023, pp. 1-6.

[5] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "Boundary-wise Loss for Medical Image Segmentation Based on Fuzzy Rough Sets", Information Sciences (Under Review)

[6] Q. Lin, X. Chen, C. Chen, N. Jathanna, S. Jamil-Copley and J.M. Garibaldi, "Study of Uncertainty of AI and Human in Cardiac MRI Segmentation", Journal of Cardiovascular Magnetic Resonance (Under Review)

Chapter 2

Background

2.1 Introduction

This thesis aims to improve the workflow of semantic segmentation from the following perspectives: 1) design a novel layer to reduce the complexity of semantic segmentation models; 2) design a boundary-wise loss function to improve the boundary accuracy of semantic segmentation; 3) design a novel quality quantification algorithm to make the semantic segmentation results more reliable and robust. Thus, Section 2.2 first introduces the background of semantic segmentation and a brief review of widely-used semantic segmentation models, semantic segmentation loss functions, and quality quantification algorithms for semantic segmentation.

Then, a number of fuzzy techniques are utilized to achieve the above aims. Fuzzy sets are applied to quantify the uncertainty, thereby indicating the segmentation quality (quality quantification algorithm). Section 2.3 focuses on the definition of various fuzzy sets: type-1 fuzzy sets and general type-2 fuzzy sets. Besides, the conditions of fuzzy union, fuzzy intersection, and fuzzy equiva-

lence relations are depicted in Section 2.3.3 to provide theoretical support for fuzzy rough sets in Section 2.4.

As fuzzy rough sets are the key point to designing the boundary-wise loss function, Section 2.4 gives a detailed description of fuzzy rough sets based on fuzzy sets, rough sets, and fuzzy equivalence relations. Section 2.5 provides the definition of fuzzy integrals and a review of commonly used fuzzy integral operators (OWA), which are utilized in semantic segmentation models to help reduce the model's complexity. Figure. 2.1 shows navigation to link each topic presented in this chapter to the aims and objectives.

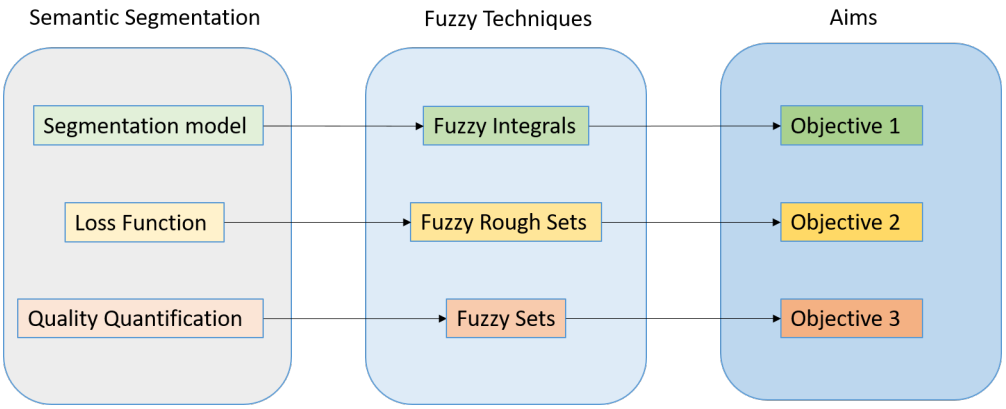


Figure 2.1: Link each topic to the aims and objectives

2.2 Semantic Segmentation

Image segmentation is the most important part of image comprehension in computer vision. It aims to divide the given image into several disjoint areas based on such extracted features as shape, grayscale value, spatial texture, and colour so that all the features share a high level of similarity in the same area. In recent years, with the rapid progress of image segmentation technology, some image-segmentation-related scenarios including object segmentation [31], human foreground segmentation [32], face parsing [33], three-dimensional reconstruction [34] have been widely used in self-driving cars, augmented reality,

security monitoring and healthcare industries.

Classical image segmentation technologies are primarily unsupervised learning methods: edge detection [9], threshold [7], region growing [8], and clustering [35]. These methods are not robust and have low-quality segmentation results when the boundary of the original image is complicated and overlapped. To mitigate this problem, end-to-end convolutional neural networks (CNNs)-based semantic segmentation techniques have gradually become the mainstream image segmentation algorithms and achieved outstanding performance in numerous computer vision tasks.

Semantic segmentation can be defined as pixel-level classification. Every individual pixel in the input image is assigned into different categories with respect to corresponding ground truth images and semantic segmentation models. Figure 2.2 shows the process of semantic segmentation. Given a raw image X, the objective of semantic segmentation is to label each pixel x_i in X with a corresponding category.

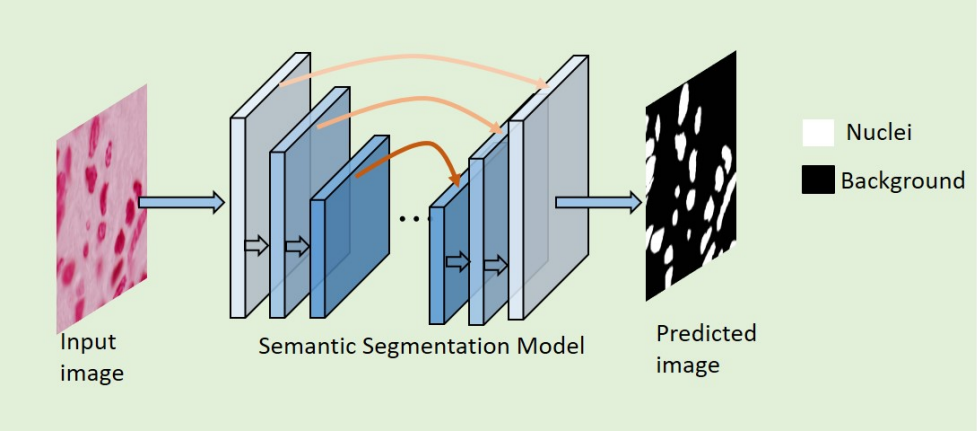


Figure 2.2: The Process of Semantic Segmentation

2.2.1 Semantic Segmentation Models

Big data and parallel computing promote the rapid development of semantic segmentation technologies. Every year there are numerous novel semantic segmentation models proposed to settle various segmentation issues. Next, some typical and widely-used semantic segmentation models is introduced.

In 2015, Long et al. [13] innovatively designed a pixel-to-pixel, end-to-end image segmentation model, Fully Convolutional Networks (FCN), based on de-convolutional operators. This model was insensitive to the details and did not take the relationship between pixels into consideration, but it was ground breaking in the semantic segmentation field. After that, many other semantic segmentation model designs and frameworks were inspired by it for instance SegNet [2], and Deconvnet [36]. In the same year, another representative semantic segmentation model, UNet, was proposed to deal with medical images [1]. To transmit learned features from the encoder to decoder directly, skip connection was applied between encoding and decoding layers in UNet. To settle 3D medical image (CT, MRI) segmentation problems, Wang et al. [37] utilized 3D convolutional kernels to substitute for 2D convolutional kernels and named the new model as 3DUNet. Furthermore, the emergence of Dilated/Atrous Convolution [15] in 2017 contributed to the production of a series of novel semantic segmentation models: DeepLab V1 [14], V2 [15], V3 [16], V3+ [17]. The most significant part of the deeplab series is that V1 used the combination of dilated convolution and Conditional Random Field (CRF), V2 added atrous special pyramid module on the basis of V1, V3 utilized the multi-grid strategy and batch normalization, and V3+ reorganised the structure of deeplab into encode-decode pattern.

Figure 2.3 and Figure 2.4 show the architectures of UNet and Segnet. Both of them have the encode-decode structure.

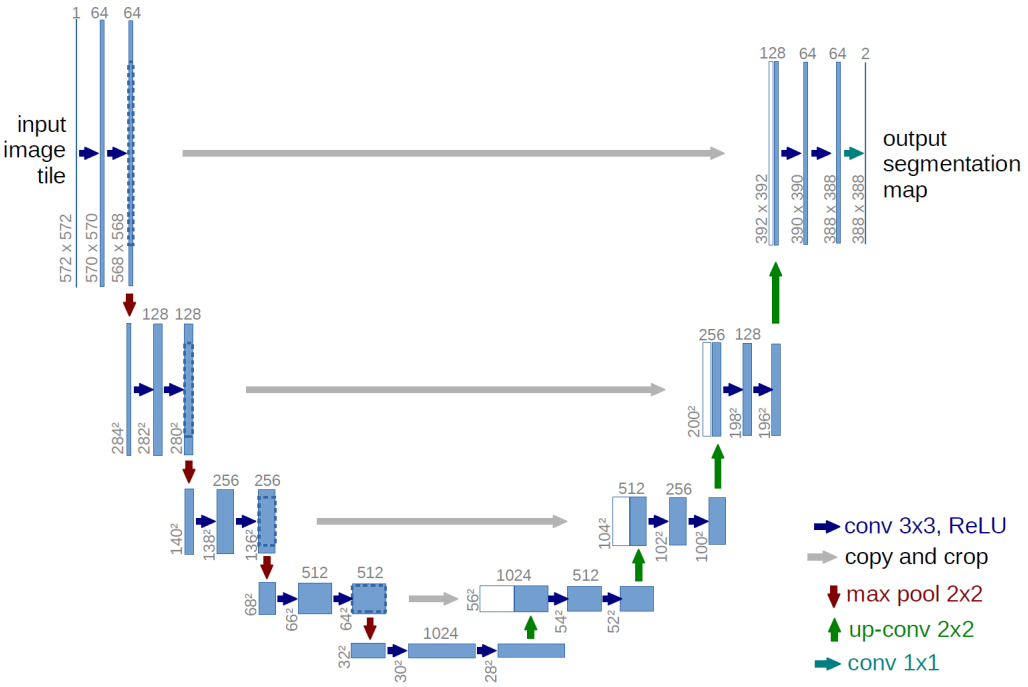


Figure 2.3: The structure of UNet model [1]

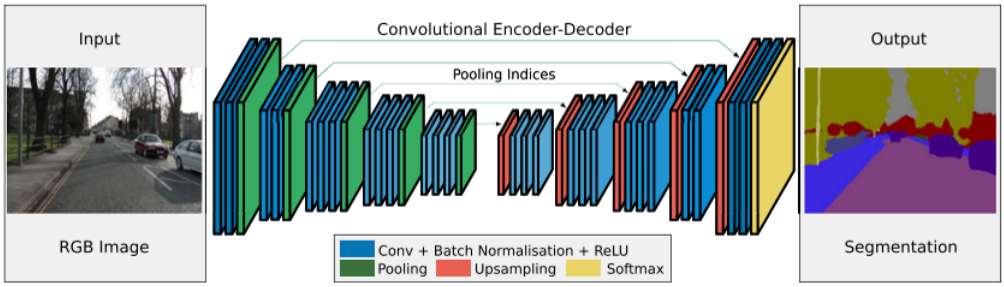


Figure 2.4: The structure of SegNet model [2]

For all the aforementioned semantic segmentation models, one of the most important components is the convolutional layer. In each convolutional layer, a convolutional kernel performs as a fusion operator, which aims to merge the spatial information and the channel information (also known as feature maps) simultaneously with respect to the corresponding learnable weights. As we know, when a model deals with multiple tasks at the same time, its performance for each individual task is potentially degraded. Thus, Hu et al. [18] demonstrated that fusing channel information separately is beneficial for the improvement of segmentation performance. In this thesis, one special fuzzy integral fusion method–OWA (Section 2.5) is leveraged to decouple the information fusion

tasks.

2.2.2 Semantic Segmentation Loss Functions

The above mentioned CNN models aim to estimate a set of intrinsic model parameters by minimizing a loss function that measures the difference between the currently predicted segmentation and the ground truth segmentation. This minimization process is performed iteratively by error back propagation using gradient descent algorithm. As the loss function guides the learning process of the semantic segmentation models, the selection of loss function is of great importance. The loss functions for semantic segmentation is divided into three categories: pixel-wise loss, region-wise loss and boundary-wise loss [38]. The pixel-wise losses include Cross Entropy Loss [19], Weighted Cross Entropy Loss [39], Balanced Cross Entropy Loss [21], and Focal Loss [22]. Cross Entropy Loss is the widely used pixel-wise loss. The others are the cross entropy variation losses, which is proposed to deal with unbalanced data issues. The Cross Entropy Loss (CELoss) is defined as:

$$L_{CE} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{K} -y_{it}\log\left(\hat{y}_{it}\right), \tag{2.1}$$

where y_{it} is the i_{th} label pixel value in the t_{th} class and $\hat{y}_{it}$ is the i_{th} predicted pixel value in the t_{th} class, K is the number of classes, N is the number of pixels.

The region-wise losses contain Dice Loss [20], Sensitivity-Specificity Loss [40], Tversky Loss [41], Focal Tversky Loss [42], Log-Cosh Dice Loss [38]. Among them, Dice Loss is the most representative and commonly used region-wise loss. DiceLoss (DLoss) is represented as:

$$L_D = \frac{1}{K}\sum_{t=1}^{K}\left(1 - \frac{2\sum_{i=1}^{N} y_{it}\hat{y}_{it}}{\sum_{i=1}^{N} y_{it}^2 + \sum_{i=1}^{N}\hat{y}_{it}^2}\right), \tag{2.2}$$

where y_{it}, $\hat{y}_{it}$, K, N have the same meaning as L_{CE}.

All the aforementioned losses are pixel-wise and region-wise losses. In semantic segmentation tasks, uncertainty and misclassification normally happen at the boundaries [26]. Therefore, if a loss function is designed to concentrate on the boundary, the image segmentation performance can be potentially improved. However, the research about boundary-wise loss is limited. There are two popular boundary-wise losses namely Hausdorff loss (HDLoss) [43] and Dual HausdorffLoss (DHDLoss) [43] proposed to focus on the segmentation boundary. HDLoss is yielded from Hausdorff distance $d(X,Y) = \max_{x \in X} \min_{y \in Y} \|x-y\|_2$, where x and y are the elements of X and Y respectively. Since the formula of Hausdorff distance is non-convex, it cannot be directly applied to calculate the image segmentation loss. The variation of Hausdorff distance has the ability to make the HDLoss tractable, which is represented as:

$$L_{HD} = \frac{1}{|\Omega|} \sum_{\Omega} \Big((p-q)^2 \otimes \left(\mathscr{D}_p^{\alpha} \right) \Big), \tag{2.3}$$

where p and q are the predicted binary image and the ground truth image respectively, $\mathscr{D}_p$ refers to the distance map of p, α is equal to 2, $\otimes$ represents the pixel-wise multiplication operator, Ω is the whole area of the given image. Since $\mathscr{D}_p$ and $\mathscr{D}_q$ are not equal, a dual direction Hausdorff loss (DHDLoss) is proposed:

$$L_{DHD} = \frac{1}{|\Omega|} \sum_{\Omega} \Big((p-q)^2 \otimes \left(\mathscr{D}_p^{\alpha} + \mathscr{D}_q^{\alpha} \right) \Big), \tag{2.4}$$

From the equation of HDLoss and DHDLoss, the distance value in the distance map has a large range which may cause instability in the training process of semantic segmentation models and lead the segmentation model to fail to converge to the optimal value. In this thesis, to manage the limitations of existing boundary-wise losses and capture the more accurate segmentation boundary, a novel boundary-wise loss function based on fuzzy rough sets (Section 2.4) is

proposed.

2.2.3 Quality Quantification for Semantic Segmentation

After being trained using the aforementioned loss function, the segmentation model can be applied to segment new images. However, in a real-world application, it is difficult to assess the semantic segmentation performance directly due to the fact that there is no ground truth image. The pre-trained semantic segmentation model can only give the predicted segmentation image and has no ability to inform the patient or clinician of the segmentation quality. Therefore, it is crucial to design a quality quantification algorithm to evaluate the segmentation performance and improve the segmentation model credibility on new data, especially in clinical settings, which is the process of quality control.

In the literature, the approaches to designing a quality quantification algorithm can be classified into three categories. The first category is registration-based quality quantification algorithms. This category generally selects some reference database with ground truth images and applies image registration between the test image and the reference images to evaluate the segmentation performance of the given test image. Valindria et al. [26] designed a reverse classifier to implement the image registration and used reverse classification accuracy to predict the segmentation performance. It is noted that registration-based quality quantification methods are time-consuming due to the fact that the image registration is implemented multiple times with numerous reference images. The second category is learning-based quality quantification algorithms. This category treats quality quantification as a regression task and utilizes various machine learning models to predict the segmentation performance. Timo et al. [44] adopted a SVM regression model based on hand-crafted features and Robinson et al. [45] applied a DCNN regression model based on the predicted image and

raw image to directly predict the Dice (a commonly-used segmentation quality metric). Instead of predicting the Dice, Shou et al. [46] proposed a generative adversarial network (GAN) model to generate the ground truth image. Then the overlap between the predicted image generated from the pre-trained segmentation model and the ground truth image generated from the GAN model was used to assess the segmentation performance.

The third category is uncertainty-based quality quantification algorithms. Unlike the previous quality quantification methods, this category leverages segmentation uncertainty to devise the quality quantification algorithm since inaccurate segmentation areas generally have high levels of uncertainty [26]. For one given test image X, these uncertainty-based methods firstly capture the data uncertainty or model uncertainty by generating N predicted images $\{y_1, y_2, y_3, \cdots, y_N\}$. Then different uncertainty measures have been proposed to quantify the uncertainty, the value of which is used to evaluate the segmentation performance. Hoebel et al. [47] and Roy et al. [48] calculated the segmentation uncertainty based on the overlap between pairwise predicted images (Dpw) and the intersection over overlap of all predicted images (IoU) respectively. These two methods Dpw and IoU were very sensitive to poor-quality predicted images, which might lead to an inaccurate evaluation of segmentation performance. Roy et al.[49] proposed a region-wise variation coefficient (VC) to measure the uncertainty. The VC was equal to pixels standard variance over pixels mean, where pixels belonged to the segmentation region of all predicted images. This method treated all the pixels in the segmentation region equally and ignored the fact that mis-segmentation commonly happened at the boundary area. Instead of directly calculating the region-wise uncertainty, Mehrtash et al. [26] and DeVries et al. [50] proposed to generate an uncertainty map firstly, in which each value referred to each pixel's uncertainty. Then Mehrtash et al. [26] computed the mean uncertainty of segmentation region (Ulabelled) and DeVries et al. [50] trained

a DCNN regression model with a raw image, predicted image and uncertainty map (CNNurp) to measure the uncertainty. These two methods Ulabelled and CNNurp did not take the influence of neighbouring pixels into consideration. Moreover, CNNurp was time-consuming and had over-fitting issues due to the fact that numerous extra parameters in the DCNN model needed to be trained compared with other uncertainty-based quality quantification methods.

Based on the above discussion, it can be concluded that uncertainty based methods have many favorable properties (e.g., computational efficiency, no learning process required, and easy to implement) compared to registration-based and learning-based methods. In this thesis, as fuzzy sets (Section 2.3) can efficiently handle ambiguity and vagueness in many fields [28, 29, 30], they are applied to quantify the segmentation uncertainty.

Next, the aforementioned fuzzy techniques (fuzzy sets, fuzzy rough sets, and fuzzy integrals) are described and discussed in detail.

2.3 Fuzzy Sets

As described in Section 2.2.3, uncertainty-based quality quantification methods are superior to registration-based and learning-based methods. Given their proven efficiency in managing ambiguity and vagueness, fuzzy sets are utilized to quantify the segmentation uncertainty. Thus, in this section, the background information for type-1 fuzzy sets, general type-2 fuzzy sets, and fuzzy set operations are given in detail.

2.3.1 Type-1 Fuzzy Sets

In 1965, Zadeh [27] first proposed the type-1 fuzzy sets to depict the ambiguity and uncertainty. Type-1 fuzzy sets are the generalizations of crisp sets. The definition of type-1 fuzzy sets is given in the following.

Definition 2.1 *Given a universe U, a type-1 (T1) fuzzy set A is a set function on U into [0,1], that is*

$$A = \{(x, \mu_A(x)) \,|x \in U\} \tag{2.5}$$

where x is a variable in the U, $\mu(x)$ *is a membership function and* $0 \leq \mu_A(x) \leq 1$. *The value of membership function refers to the degree of x belonging to A.*

When the universe U is discrete, the type-1 fuzzy set A is normally expressed as

$$A = \sum_{i=1}^{N} \frac{\mu_A(x_i)}{x_i} \tag{2.6}$$

where N is the number of discrete points in the U and x_i is the i_{th} discrete value. Note that the $\sum$ means the set of all discrete points x_i with its corresponding membership value $\mu_A(x_i)$ instead of the summation operator.

When the universe U is continuous, the type-1 fuzzy set A is commonly represented by

$$A = \int_{x \in U} \frac{\mu_A(x)}{x} \tag{2.7}$$

where the $\int$ refers to the set of all $x \in U$ with its corresponding membership value $\mu_A(x)$ instead of the integral operation.

Figure 2.5 shows the examples for two type-1 fuzzy sets.

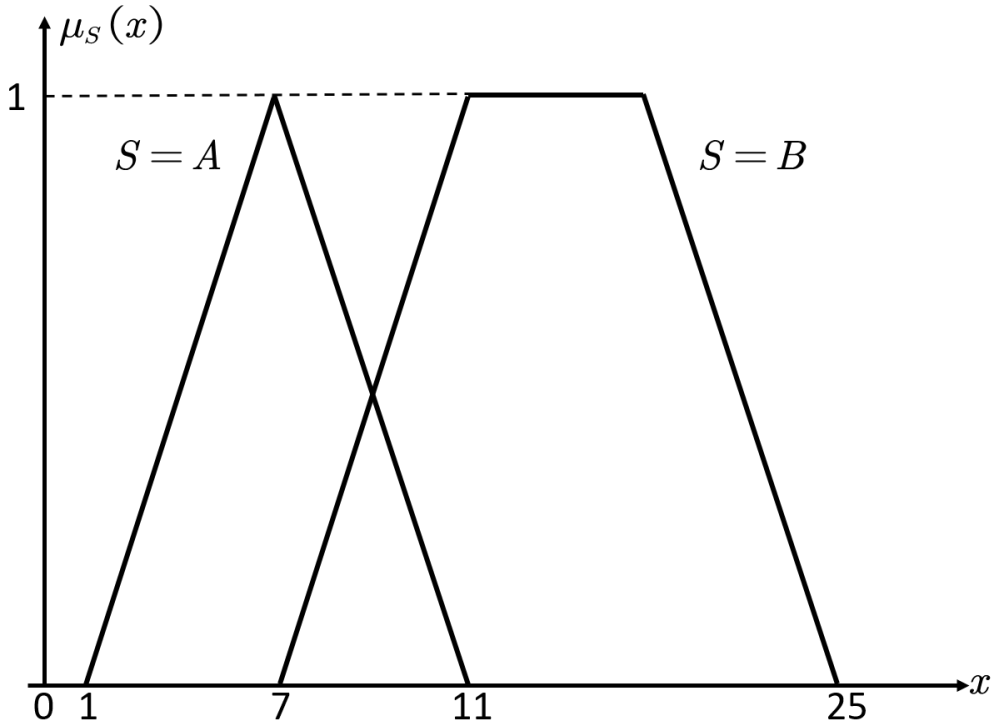


Figure 2.5: Example of type-1 fuzzy sets [3]

2.3.2 Type-2 Fuzzy Sets

Type-2 fuzzy sets are the generalization of type-1 fuzzy sets in order to handle more uncertainty. Although type-1 fuzzy sets have been applied to represent numerous complicated real-world problems [51, 52], criticism were made by many researchers [53] about the fact that the type-1 fuzzy sets do not include the uncertainty of their membership grade. To overcome the limitation of type-1 fuzzy sets, Zadeh [54] first proposed the concept of type-2 fuzzy sets to incorporate uncertainty about the membership grade. Then Mendel and John promoted the development of type-2 fuzzy sets by the introduction of footprint of uncertainty (FOU). FOU makes type-2 fuzzy sets visual and straightforward to understand. Currently, type-2 fuzzy sets are widely-used in various areas: Business and Management [55], Environmental Sciences [56], Computer Science [57], Engineering [58] etc. The definition of type-2 fuzzy sets is expressed as:

Definition 2.2 *Given a universe U, a type-2 (T2) fuzzy set $\tilde{A}$ can be treated as the extension of a type-1 fuzzy set and is represented by*

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) | x \in U, u \in J_x \subseteq [0,1]\}. \tag{2.8}$$

where x is the primary variable and u is the secondary variable. $\mu_{\tilde{A}}(x,u)$ is the secondary membership function and the value of $\mu_{\tilde{A}}(x,u)$ is the secondary membership grade. J_x denotes the primary membership.

In the type-2 fuzzy set $\tilde{A}$, given a variable x, the related membership grade of x in $\tilde{A}$ is a function $\mu_{\tilde{A}}(x,u)$ instead of a single value. This function includes the uncertainty of the membership grade, which is the key point of type-2 fuzzy sets.

When the universe U is discrete, the type-2 fuzzy set $\tilde{A}$ is normally expressed as

$$\tilde{A} = \sum_{x \in U} \sum_{u \in J_x} \frac{\mu_{\tilde{A}}(x,u)}{(x,u)} \quad J_x \in [0,1] \tag{2.9}$$

where $\sum$ means the collection of all discrete points x with its corresponding membership function $\mu_{\tilde{A}}(x,u)$ instead of the summation operator.

When the universe U is continuous, the type-2 fuzzy set $\tilde{A}$ is commonly represented by

$$\tilde{A} = \int_{x \in U} \int_{u \in J_x} \frac{\mu_{\tilde{A}}(x,u)}{(x,u)} \quad J_x \in [0,1] \tag{2.10}$$

where the $\int$ denotes the collection of all $x \in U$ with its corresponding membership function $\mu_{\tilde{A}}(x,u)$ instead of the integral operation. Figure 2.6 shows an example for general type-2 fuzzy sets.

Despite the fact that general type-2 fuzzy sets have been successfully used in many industries, it has a significant computational cost for handling uncertainty. To improve the computational efficiency, in practical application, a special case

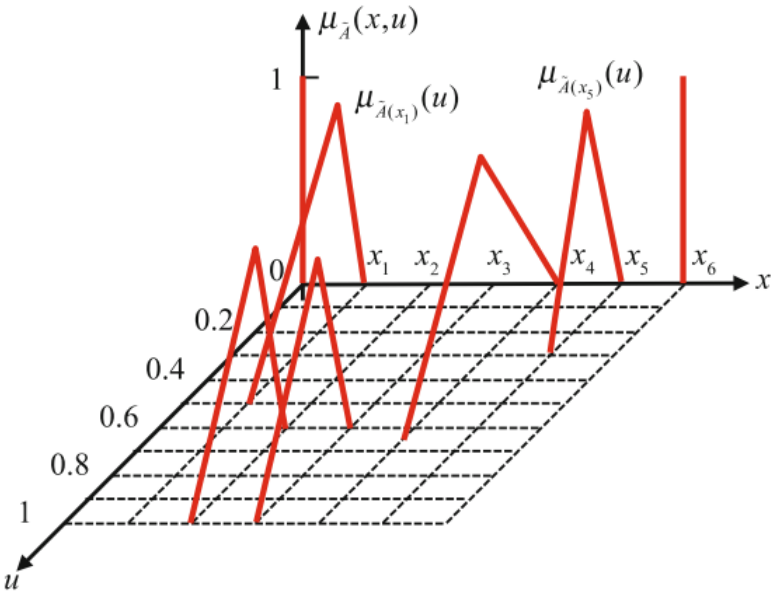


Figure 2.6: Example of type-2 fuzzy sets [4]

of type-2 fuzzy sets namely interval fuzzy sets are commonly used since the interval fuzzy sets use the interval to depict the uncertainty of the membership grade and straightforward to understand.

Definition 2.3 *When the secondary membership grades* $\mu_{\tilde{A}}(x,u) = 1$*, the general type-2 fuzzy sets are simplified to interval fuzzy sets and be expressed as:*

$$\tilde{A} = \int_{x \in U} \int_{u \in J_x} \frac{1}{(x,u)} \quad J_x \in [0,1] \tag{2.11}$$

where the universe U *is continuous, and*

$$\tilde{A} = \sum_{x \in U} \sum_{u \in J_x} \frac{1}{(x,u)} \quad J_x \in [0,1] \tag{2.12}$$

where the universe U *is discrete.*

Figure 2.7 visualizes the interval fuzzy sets. From Figure 2.7, the interval fuzzy sets can be represented by upper membership function (UMF) $\bar{\mu}_{\tilde{A}}(x)$ and lower

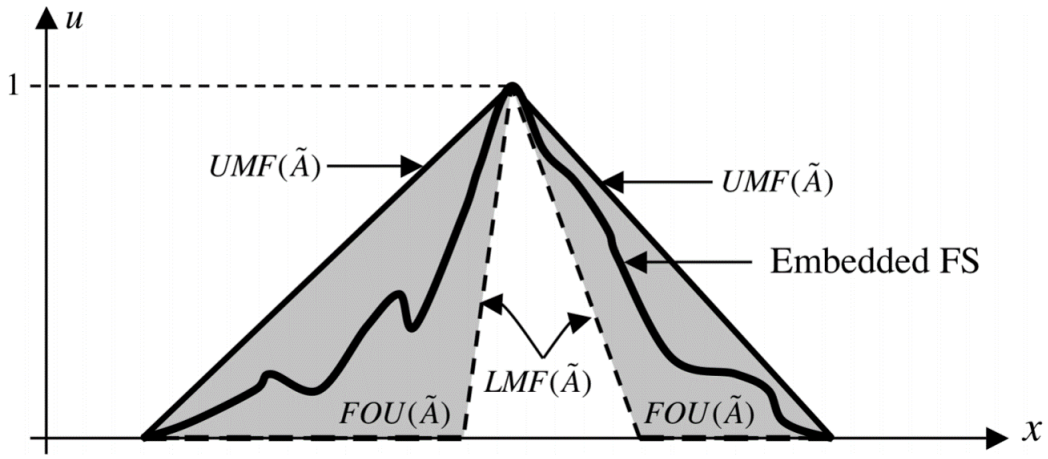


Figure 2.7: Example of interval fuzzy sets [5]. The solid line is the upper boundary of FOU and refers to the upper membership function (UMF). The dotted line is the lower boundary of FOU and represents the lower membership function (LMF). The shaded area is the FOU.

membership function (LMF) $\underline{\mu}_{\tilde{A}}(x)$, that is

$$\tilde{A} = \left\{ \left(x, \left[\underline{\mu}_{\tilde{A}}(x), \bar{\mu}_{\tilde{A}}(x) \right] \right) | x \in U \right\} \tag{2.13}$$

where $\underline{\mu}_{\tilde{A}}(x) = \{x, \min(J_x)\}$ and $\bar{\mu}_{\tilde{A}}(x) = \{x, \max(J_x)\}$, J_x is the primary membership grade(s) of $\tilde{A}$ at x.

2.3.3 Fuzzy Set Operations

As the same with conventional set theory, the fuzzy sets also have union, intersection, and complement operators. However, the fuzzy sets operators are not restricted strictly in one definite form. Provided an operator satisfies some essential conditions, it can be reasonably applied in fuzzy sets.

Fuzzy union is defined as s-norm, that is $S[\mu_A(x), \mu_B(x)] = \mu_{A \cup B}(x)$, where $S : [0,1] \times [0,1] \to [0,1]$ and have the following attributes:

- $S(1,1) = 1, S(0,a) = S(a,0) = a$ (boundary condition);

- $S(a,b) = S(b,a)$ (commutativity);
- if $a \leq a'$ and $b \leq b'$, $S(a,b) \leq S(a',b')$ (monotonicity);
- $S(S(a,b),c) = S(a,S(b,c))$ (associativity).

Fuzzy intersection is given as t-norm $T[\mu_A(x), \mu_B(x)] = \mu_{A\cap B}(x)$, which also satisfies the commutativity, monotonicity, and associativity like s-norm:

- $T(a,1) = a, T(1,a) = a$ (boundary condition);
- $T(a,b) = T(b,a)$ (commutativity);
- if $a \leq a'$ and $b \leq b'$, $T(a,b) \leq T(a',b')$ (monotonicity);
- $T(T(a,b),c) = T(a,T(b,c))$ (associativity).

Fuzzy complement N is a decreasing map $N(\mu_A(x)) = 1 - \mu_A(x)$ where $N : [0,1] \to [1,0]$.

A fuzzy equivalence relation is a function $D : X^2 \to [0,1]$ and satisfy the following conditions:

- $D(x,x) = 1$ (reflexivity);
- $D(x,y) = D(y,x)$ (symmetry);
- $\min(D(x,y), D(y,z)) \leq D(x,z)$ (min-max-transitivity)

It is noted that 'min' is the special case of t-norm. Thus, when the operator 'min' is replaced by t-norm, that is $T(D(x,y), D(y,z)) \leq D(x,z)$, D is called a fuzzy T-equivalence relation.

Table 2.1 and Table 2.2 show the widely used s-norm and t-norm operators satisfying the corresponding conditions.

Table 2.1: Examples for s-norm operators

s-norm
$S_M(a,b) = \max(a,b)$
$S_P(a,b) = a+b-ab$
$S_L(a,b) = \min(a+b,1)$
$S_{\cos}(a,b) = \min\left(a+b-ab+\sqrt{2a-a^2}\sqrt{2b-b^2},1\right)$

Table 2.2: Examples for t-norm operators

t-norm
$T_M(a,b) = \min(a,b)$
$T_P(a,b) = a \times b$
$T_L(a,b) = \max(a+b-1,0)$
$T_{\cos}(a,b) = \max\left(ab-\sqrt{1-a^2}\sqrt{1-b^2},0\right)$

2.4 Fuzzy Rough Sets

As described in Section 2.2.2, the application of the boundary-wise loss function in semantic segmentation models benefits the improvement of the boundary accuracy of semantic segmentation. Fuzzy rough sets are normally utilized to measure the relationship between two sets using fuzzy equivalence relation. Thus, using fuzzy rough sets to calculate the difference between the boundary sets of the predicted image and the ground truth image is a significant direction of the investigation. In this section, the definition of fuzzy rough sets and their corresponding generalization are presented in detail.

2.4.1 Rough Sets

Rough sets [59] are particularly useful in dealing with ambiguity, vagueness and general uncertainty problems. Let $IS = \langle U, C \rangle$ be an information system, where U is a nonempty and finite set of objects, C is a nonempty and finite set

of attributes. Given an equivalence relation R, the universe U is divided into a family of disjoint subsets, named equivalence class $[x]_R$. For $\forall x,y,z \in U$, the equivalence relation R is satisfied the following rules: $R(x,x) = 1$, $R(x,y) = R(y,x)$, $R(x,y) = 1, R(y,z) = 1 \Longrightarrow R(x,z) = 1$.

According to [59], given a subset of objects $X \subseteq U$ and an equivalence relation R, the lower and upper approximations of X are defined as:

$$\begin{cases} \underline{R}X = \{[x]_R \,|\, [x]_R \subseteq X\} \\ \overline{R}X = \{[x]_R \,|\, [x]_R \cap X \neq \oslash\} \end{cases} \tag{2.14}$$

When $\underline{R}X \neq \overline{R}X$, X is a rough set in the approximation space. Obviously, the subset X is approximated by two unions of equivalent classes. The lower approximation of X is represented by the union of equivalence classes $[x]_R$ which are totally contained by X. The upper approximation of X is evaluated by the union of equivalence classes which has a non-empty intersection with X. The difference between $\underline{R}X$ and $\overline{R}X$ is the boundary region.

2.4.2 Fuzzy Rough Sets

Although the classical rough sets have been applied in many fields and also have obtained outstanding performance, there is a restriction that could not be ignored. The classical rough sets are considerably effective only when the data is symbolic-valued [60]. To address the above restriction and broaden the application range of rough sets, Dubois et al. [24] imposed a novel theory named fuzzy rough sets which was integrated with fuzzy sets and rough sets. The key idea of the fuzzy rough sets is to supersede the equivalence relation of rough sets by a fuzzy equivalence relation.

Definition 2.4 *Given a universe U, R is a fuzzy equivalence relation on U. For $\forall x, y \in U$, the fuzzy rough sets are defined as [24]*

$$\begin{cases} \underline{R_{\max}} X(x) = \underset{y \in U}{inf} \max\left(1 - R(x,y), X(y)\right) \\ \overline{R_{\min}} X(x) = \underset{y \in U}{sup} \min\left(R(x,y), X(y)\right) \end{cases} \tag{2.15}$$

where X is a subset of U.

Based on [61][62], if the R is a fuzzy T-equivalence relation on U, the more general fuzzy operator t-norm, s-norm defined above can be applied to substitute 'max' and 'min', that is

$$\begin{cases} \underline{R_s} X(x) = \underset{y \in U}{inf\, s}\left(N\left(R(x,y)\right), X(y)\right) \\ \overline{R_t} X(x) = \underset{y \in U}{sup\, t}\left(R(x,y), X(y)\right) \end{cases} \tag{2.16}$$

where X is the subset of U and N is the fuzzy complement. $\underline{R_{\max}} X(x)$ and $\underline{R_s} X(x)$ are the lower approximation and mean the degrees the x certainly belongs to the set X. $\overline{R_{\min}} X(x)$ and $\overline{R_t} X(x)$ are the upper approximation and denote the degrees the x possibly belongs to set X.

2.5 Fuzzy Integral

As mentioned in Section 2.2.1, fusing channel information separately is beneficial for the improvement of segmentation performance. Fuzzy integrals are an effective method for fusing information, and some special fuzzy integral operators (OWA) are free of parameters and easy to implement in deep-learning models. Thus the special fuzzy integral fusion method—OWA is leveraged to decouple the information fusion tasks. In this section, the definition of fuzzy integrals and the detailed process of OWA operators are presented.

Fuzzy integrals are a set of non-linear aggregation operators calculated by the associated fuzzy measures. The fuzzy measure is a monotonic increasing set function $\rho : 2^X \to \mathbb{R}^+$(where $X = \{x_1, x_2, ..., x_n\}$ refers to an infinite set) and has the following attributes: (1) boundary condition: $\rho(\phi) = 0, \rho(X) = 1$; (2) monotonicity: if $A \subseteq B$ and $A, B \subseteq X, \rho(A) \leq \rho(B)$. As fuzzy integrals have the ability to model interaction between different information sources, they are superior to other fusion algorithms. Choquet integrals [63] are the widely used fuzzy integrals and defined as:

$$\int h \circ \rho = C_\rho(\theta) = \sum_{i=1}^{N} \left[\theta\left(x_{\sigma(i)}\right) - \theta\left(x_{\sigma(i+1)}\right)\right] \rho(A_i), \tag{2.17}$$

where θ is a vector and has a value for each attribute in X ($x_{\sigma(i)} \in X$), $A_i = x_{\sigma(1)}, x_{\sigma(2)}, \cdots, x_{\sigma(i)}$, σ is the permutation $\theta\left(x_{\sigma(1)}\right) \geq \theta\left(x_{\sigma(2)}\right) \geq \cdots \geq \theta\left(x_{\sigma(N)}\right)$, ρ is fuzzy measures.

It is noted that the Choquet integral is a kind of flexible aggregation algorithm and different types of fuzzy measures enable the Choquet integral to generate some specific fusion methods. Yager [64] investigated the connection between Choquet integral with ordered weighted averaging (OWA) and concluded that when the fuzzy measure belonged to the cardinality-based measure, Choquet integral could be regarded as OWA operator. In the next subsection, the special case of Choquet integral—OWA is described in detail.

2.5.1 OWA Operators

Ordered weight averaging (OWA) operators were innovatively introduced for information fusion by Yager [64]. An OWA operator has the ability of mapping n-dimension inputs to 1-dimension output: $\mathbb{R}^n \to \mathbb{R}$. The details about OWA are given below.

Definition 2.5 *Given a set of values* $\sigma_1, \sigma_2, ..., \sigma_n$*, weights vector* $\omega = [\omega_1, \omega_2, ..., \omega_n]^T$*, where* $\forall \omega_i \in [0,1], i = 1,2,...,n$ *and* $\sum_{i=1}^{n} \omega_i = 1$*, the OWA operator can be defined as:*

$$f_{OWA}(\sigma_1, \sigma_2, ..., \sigma_n) = \sum_{i=1}^{n} \omega_i \sigma_{(i)}. \tag{2.18}$$

where $\sigma_{(i)}$ *is the* i_{th} *largest element of the n-dimensional inputs* $\sigma_1, \sigma_2, ..., \sigma_n$..

According to the definition of OWA, three steps are required to calculate the final aggregating value:

- Step1: sort the feature maps $\sigma_1, \sigma_2, ..., \sigma_n$ in descending order;
- Step2: design an appropriate algorithm to obtain the optimal OWA weights vector $\omega = [\omega_1, \omega_2, ..., \omega_n]^T$;
- Step3: aggregate the reorder feature maps with OWA weights by equation (**??**).

Obviously, how to calculate the weights vector plays a key role in the OWA operator. Herein, some existing methods with respect to OWA weights are succinctly reviewed.

Firstly, linguistic quantifier Q [65] was applied to search for OWA weights:

$$\omega_i = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right), i = 1,2,...,n. \tag{2.19}$$

If linguistic quantifier Q means 'For all', the OWA weights can be found by

$$\begin{aligned} Q(k) &= \begin{cases} 0, for\, k < 1 \\ 1, for\, k = 1 \end{cases} \\ \omega_i &= \begin{cases} 0, i < n \\ 1, i = n \end{cases} \end{aligned} \tag{2.20}$$

If linguistic quantifier Q represents 'Identity', the OWA weights are evaluated via

$$\begin{aligned} Q(k) &= k \\ \omega_i &= \frac{1}{n}, i = 1,2,...,n \end{aligned} \tag{2.21}$$

Then, O'Hagan [66] designed a novel algorithm, which maximizes the dispersion measure [64] and predefines the orness measure value [64], to capture the OWA weights:

$$\begin{aligned} &Max \sum_{i=1}^{n} \omega_i \ln \omega_i \\ subject\ to\ &\frac{1}{n-1} \sum_{i=1}^{n} (n-i)\, \omega_i = \varepsilon, \\ &\sum_{i=1}^{n} \omega_i = 1 \end{aligned} \tag{2.22}$$

where $0 \leqslant \varepsilon \leqslant 1$ is the orness value and $i = 1,2,...,n$. Furthermore, Yager and Filev [67] investigated some widely-used families of OWA weights.

(1)

$$\omega_i = \begin{cases} \frac{1}{n}(1-\upsilon) + \upsilon, i = 1 \\ \frac{1}{n}(1-\upsilon), i = 2,...,n \end{cases} \tag{2.23}$$

where $0 \leqslant \upsilon \leqslant 1$

(2)

$$\omega_i = \frac{\beta_i^{\tau}}{\sum_{i=1}^{n} \beta_i^{\tau}} \tag{2.24}$$

where $-\infty \leqslant \tau \leqslant +\infty$, β_i is the i largest element of the n-dimension inputs of the aggregation operator.

(3)

$$\begin{aligned} &\omega_1 = \upsilon, \omega_2 = \upsilon(1-\upsilon), \omega_3 = \upsilon(1-\upsilon)^2, \\ &..., \omega_{n-1} = \upsilon(1-\upsilon)^{n-2}, \omega_n = (1-\upsilon)^{n-1} \end{aligned} \tag{2.25}$$

(4)

$$\omega_1 = \upsilon^{n-1}, \omega_2 = (1-\upsilon)\,\upsilon^{n-2}, \omega_3 = (1-\upsilon)\,\upsilon^{n-3},$$
$$..., \omega_{n-1} = (1-\upsilon)\,\upsilon, \omega_n = (1-\upsilon) \tag{2.26}$$

Lastly, normal distribution was introduced to generate OWA weights [68]:

$$\omega_i = \frac{1}{\sqrt{2\pi\sigma_n}} e^{-\left[\frac{(i-\mu_n)^2}{2\sigma_n^2}\right]} \tag{2.27}$$

where $\mu_n = \frac{1+n}{2}, \sigma_n = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(i-\mu_n)^2}$.

2.6 Summary

This chapter provides detailed background material and an overview of the literature this thesis uses and refers to.

As shown in Figure 2.1, the first aspect is to incorporate fuzzy integral to the semantic segmentation model, which creates a modeling framework in which the number of parameters is impressively low (objective 1). The architectures and characteristics of the past and current semantic segmentation models are reviewed. To overcome one of these models' limitations which extracts and aggregates spatial information and channel-wise features simultaneously, the fuzzy integral is utilized to fuse the channel information separately. The definitions and formulas of different fuzzy integral operators are discussed in detail.

The second aspect is to calculate the boundary-wise semantic segmentation loss using fuzzy rough sets, which improves overall segmentation accuracy with particular emphasis on boundaries (objective 2). A brief review of semantic segmentation loss functions and their corresponding challenges are provided. The

boundary-wise loss function is to calculate the difference between the boundary sets of the predicted image and the ground truth image. It indicates that the boundary-wise loss pays more attention to the boundary compared to other losses. Fuzzy rough sets are normally utilized to measure the relationship between two sets using fuzzy equivalence relation and can be utilized to design the novel boundary-wise loss. Some basic theoretical knowledge and definitions of fuzzy rough sets are introduced, which constructs a solid mathematical support for the derivation procedure of the boundary-wise loss function.

The third aspect is to leverage fuzzy sets to evaluate the semantic segmentation quality namely quality quantification algorithms, which makes semantic segmentation results more reliable and robust (objective 3). The literature about designing various quality quantification algorithms is reviewed and the advantages and disadvantages of different kinds of quality quantification algorithms are also highlighted. Uncertainty-based quality quantification methods are superior to registration-based and learning-based methods. Given their proven efficiency in managing ambiguity and vagueness, fuzzy sets are utilized to quantify the segmentation uncertainty. The definition and representation methods of fuzzy sets are described in detail since they play a significant role in the proposed new fuzzy-based quality quantification algorithm.

With the background information, in the next chapter, a novel fuzzy integral layer based on some special fuzzy integral operators–OWA is proposed, which can be inserted into many semantic segmentation models and help reduce the number of parameters.

Chapter 3

A Novel Fuzzy Integral Module in Semantic Segmentation Models

The most crucial and fundamental component of semantic segmentation is segmentation models. As mentioned in Section 2.2.1, the widely-used semantic segmentation models like FCN, SegNet, UNet, DeepLab, etc. are constituted by convolutional neural networks (CNNs). CNNs normally extract and aggregate spatial information and channel-wise features simultaneously. In order to achieve promising segmentation performance, it is required to involve numerous learnable parameters, which increase the model's complexity. Moreover, when a model deals with multiple tasks at the same time, its performance for each individual task is potentially degraded. Thus, it is a natural idea to decouple the spatial and channel information fusion tasks in semantic segmentation models. Most studies have concentrated on spatial information by modifying convolutional kernel size, while channel information has received less attention. Note that fuzzy integrals are an effective method for fusing information, and some special fuzzy integral operators are free of parameters. Applying the fuzzy integral to fuse the channel information can address objective 1 in Section 1.2.

Therefore, in this chapter, as presented in our study [69], a novel fuzzy integral module is introduced to the CNNs for fusing the information across feature channels.

Section 3.1 introduces the motivation and the importance of channel-wise features in semantic segmentation. In detail, the framework of the proposed fuzzy integral module and the corresponding fusion operators are described in Section 3.2. In Section 3.3, two experiments are conducted: one is to compare with other fuzzy-integral-based semantic segmentation models, and the other is to apply the proposed fuzzy integral module in a widely-used semantic segmentation model to verify its generalizability. Section 3.4 further discusses the benefits and restrictions of our suggested methodology. The contributions and conclusion of this chapter are summarized in Section 3.5.

3.1 Background and Motivation

Convolutional neural networks (CNNs), as one of the most successful deep learning (DL) architectures, have been commonly used in semantic segmentation tasks. One of the most important components in CNNs is convolutional layer. In each convolutional layer, a convolutional kernel performs as a fusion operator, which aims to merge the spatial information and the channel information (also known as feature maps) with respect to the corresponding learnable weights. Spatial information refers to the location relationship of different pixels in the same image or the feature map extracted from the previous CNN layer. In CNNs, the convolutional filter sizes contribute to capturing the spatial information in different resolutions. Therefore, the selection of appropriate filters plays an important role in model performance and it has drawn increasing research attention. Szegedy et al. [70] designed an inception module including

two different convolutional kernels 3×3 and 5×5 which enabled multi-scale spatial information to be captured, leading to improved performance. Newell et al. [71] built on their work for human pose estimation by utilizing stacked hourglass networks. This network module allows the extraction of local features in multiple scales (e.g. face and hands), followed by an integration of these features to derive the full-body pose.

Channel information represents different feature values at the same image location. The conventional CNN model simply adds the feature values of all channels to form the feature maps of the consecutive layer. By adding squeeze-and-excitation (SE) blocks, Hu et al. [18] trained a neural network that is able to capture the relationships between feature maps. By combining the proposed SE blocks and CNNs, this framework won first place at ILSVRC 2017. However, it is well-known that CNN can be considered as a black box. Although it can be used to achieve high performance in many applications, the black box characteristic makes it difficult to explore the best feature map fusion approach and the related studies are very few.

On the other hand, as described in Section 2.5, fuzzy integrals originally introduced by Sugeno [72] are effective information fusion technologies. The most commonly used fuzzy integrals are Choquet integral [63] and Sugeno integral [72], which are both calculated with respect to relevant fuzzy measures. Fuzzy integrals are a family of flexible and interpretable aggregation operators and have been widely applied in artificial intelligence. Single Choquet integral classifier and cross-oriented Choquet integral classifier were designed to handle classification tasks in pattern recognition [73]. Hadjadji and Chibani [74] devised a novel writer identification system by combining three texture descriptors: run length (RL), oriented Basic Image Feature (oBIF) and Local phase quantization(LPQ) based Choquet integral. However, the application of fuzzy integrals in deep learning is still limited. Most of the published literature [75, 76]

has focused on the fusion of different CNN models by fuzzy integrals, which normally consists of choosing more than one model for the same dataset and utilizing fuzzy integrals to fuse the outputs of different models. Although the fusion of CNN models by fuzzy integrals has achieved outstanding performance in many CV tasks, very few studies have explored the use of fuzzy integrals to aggregate at the feature level, meaning that fuzzy integrals are introduced in the hidden layers of deep learning models rather than the output layer.

To the best of our knowledge, only Price et al. [6] have explored feature fusion by fuzzy integrals in deep learning models. In their method, the feature maps were first sorted in descending order according to their image entropy values and then the top five feature maps were selected to be fused via fuzzy integrals. It was the first study that fuzzy integrals had been introduced to aggregate deep features. This idea is novel and attractive but some drawbacks have been identified as follows. One drawback is that most model weights remain unchanged since only retaining the top five and discarding the other feature maps have a negative effect on the loss propagation process. The other drawback is that the entropy-based sorting algorithm is pixel-wise and extremely time-consuming, hence not suitable for practical applications. To address these limitations, a new fuzzy module that includes an additional convolutional layer for feature map dimensionality reduction and a Choquet integral module for information fusion across feature channels is proposed. The proposed fuzzy integral module addresses objective 1 in this thesis, and the next section provides a detailed explanation of how it works.

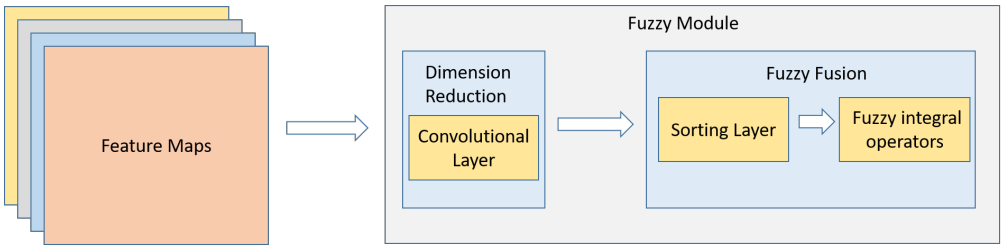


Figure 3.1: The structure of the fuzzy integral module

3.2 Fuzzy Integral Module

In this section, a new fuzzy module is proposed to combine channel information of CNNs which includes a dimensionality reduction layer and Choquet integral fusion operators.

Figure 3.1 represents the details of the proposed fuzzy module consisting of a dimensionality reduction layer and a fuzzy fusion layer. The dimensionality reduction layer is a convolutional operator, which helps reduce the number of feature maps. This operator guarantees the success of loss back propagation as well as keeping all weights updated regularly. The fuzzy fusion layer comprises Choquet integral fusion operators that are executed to generate new feature maps, where each fusion operator derives one feature map. As mentioned in Section 2.5, when the fuzzy measure belonged to the cardinality-based measure, the Choquet integral could be regarded as OWA operator which has a straightforward computation process. Thus, in this chapter, the special cases of Choquet integral —OWA operators are chosen to integrate the input feature maps. An OWA operator has the ability to map n-dimensional inputs to 1-dimensional output and the definition is given as follows.

Definition 3.1 *Given a set of values* $\sigma_1, \sigma_2, ..., \sigma_n$*, weights vector* $\omega = [\omega_1, \omega_2, ..., \omega_n]^T$*, where* $\forall \omega_i \in [0,1], i = 1,2,...,n$ *and* $\sum_{i=1}^{n} \omega_i = 1$*, the OWA operator can be de-*

fined as:

$$f_{OWA}(\sigma_1, \sigma_2, ..., \sigma_n) = \sum_{i=1}^{n} \omega_i \sigma_{(i)}. \tag{3.1}$$

where $\sigma_{(i)}$ is the i_{th} largest element of the n-dimensional inputs $\sigma_1, \sigma_2, ..., \sigma_n$.

According to the definition of OWA, three steps are followed to calculate the final aggregated value:

- Step1: sort the feature maps $\sigma_1, \sigma_2, ..., \sigma_n$ in descending order according to a predefined evaluation metric.
- Step2: design an appropriate algorithm to obtain the optimal OWA weights vector $\omega = [\omega_1, \omega_2, ..., \omega_n]^T$. Each ω is associated with one feature map.
- Step3: aggregate the reordered feature maps with OWA weights by equation (3.1).

For step 1, as each feature map refers to a single channel image, image quality evaluation indices, e.g. the Laplacian gradient function, and Entropy function [77] can be applied to sort the feature maps. For step 2, Section 2.5.1 describes the most frequently used OWA weight vectors.

One OWA aggregation operator refers to one kind of relationship between feature maps and the application of multiple fusion operators can contribute to enhancing the utilization of feature maps. N_f OWA operators generate N_f new feature maps being the inputs of the subsequent CNN layers. The conventional convolutional layer aggregates the feature maps in the spatial dimension by the corresponding weights learned from training data, while the OWA operators fuse feature maps in the channel dimension by predefined weights based on the image quality of each feature map. Therefore, adding the proposed fuzzy module into the baseline deep CNN model is beneficial for achieving feature fusion along both spatial and channel dimensions. Furthermore, the fuzzy module is

flexible and can be inserted after convolutional layers, max-pooling layers, deconvolutional layers, or activation function layers. The fuzzy integral fusion operators also have numerous options.

3.3 Evaluation and Results

As mentioned in Section 3.1, only Price et al. [6] have utilized fuzzy integrals to fuse feature maps in the intermediate layers of CNN models. Therefore, in this section, the first experiment is to compare the proposed method with Price's method [6]. Then, to investigate the performance of the proposed fuzzy integrals module in widely-used semantic segmentation models, the second experiment is to apply the fuzzy integral module to UNet [1]. Three public datasets for image segmentation have been used in both experiments.

- ISIC-2018: a dataset of dermoscopic lesion images which aims to automatically segment skin lesions and extract corresponding boundaries. It has 2594 dermoscopic images with corresponding masks annotated by experts.
- The second dataset comes from the Kaggle Carvana Image Masking Challenge Task (https://www.kag-gle.com/c/carvana-image-masking-challenge), the goal of which is to segment cars from a given image. It includes 5088 images with corresponding binary label images.
- The third dataset is derived from the 2018 Data Science Bowl which aims to detect the nuclei edges in divergent images to boost medical discovery (https://www.kaggle.com/c/data-science-bowl-2018-/data). It consists of 670 images generated from various conditions: their imaging modality, magnification and cell types also differ from one another. The correspond-

ing binary mask for each image is also available.

Implementation detail and experimental results are given as follows.

3.3.1 Implementation Details

Firstly, the three public datasets were divided into training datasets and testing datasets by the ratio of 8:2. Then the training datasets were applied to train the baseline model, and the testing datasets were utilized to evaluate the performance. After that, the proposed fuzzy module represented in Figure 3.1 was implemented to enhance the segmentation performance. For the proposed fuzzy module, a convolutional layer with five filters was applied as the dimensionality reduction operator. It should be emphasized that the filter quantity can be adjusted according to the complexity of the application scenarios. We used five for the purpose of a fair comparison with Price's method, as they retained 5 feature maps and discarded others.

Six mostly commonly used OWA operators were utilized for fuzzy fusion and their corresponding weights were: a) $\omega = (1,0,...,0)$, b) $\omega = (0,...,0,1)$, c) $\omega = (\frac{1}{n},...,\frac{1}{n},...,\frac{1}{n})$, d) $\omega = \left(0,...,0,\frac{1}{2},\frac{1}{2},0,...,0\right)$, e) $\omega_1 = \upsilon, \omega_2 = \upsilon(1-\upsilon), \omega_3 = \upsilon(1-\upsilon)^2, ..., \omega_{n-1} = \upsilon(1-\upsilon)^{n-2}, \omega_n = (1-\upsilon)^{n-1}$, f) $\omega_1 = \upsilon^{n-1}, \omega_2 = (1-\upsilon)\upsilon^{n-2}, \omega_3 = (1-\upsilon)\upsilon^{n-3}, ..., \omega_{n-1} = (1-\upsilon)\upsilon, \omega_n = (1-\upsilon)$, where $n = 5, \upsilon = 0.5$. On the basis of OWA definition, before aggregating the feature maps, Laplacian gradient function was applied to sort them in descending order. Compared to the entropy-based image quality evaluation index in [6], the Laplacian method is more efficient since it calculates using a 3×3 sliding window rather than pixel-wise. Then the reordered feature maps with OWA weights were aggregated by equation (3.1). Six OWA operators represented six different combinations for feature maps, which enhanced the utilization of information.

All of the experiments were conducted on a computer with the following specification: i7-3820 CPU and NVIDIA GeForce GTX1080Ti. The networks were implemented using Pytorch and were trained 100 epochs with a learning rate of 10^{-4} using Adam optimization method.

3.3.2 Comparison with Other Fuzzy-integral-based Semantic Segmentation Model

In order to make a fair comparison with Price's fuzzy model [6], the same model architecture is selected as our baseline model in this experiment (shown in Figure 3.2 (a)). It comprises two parts: down-sampling process and up-sampling process. Four convolutional layers with 5×5 kernels (white arrows) and two maxpooling layers (black arrows) constitute the down-sampling strategy. The up-sampling procedure consists of two deconvolutional layers with 6×6 kernels (blue arrows). The activation function is relu function which is expressed as $f(x) = \max(0, x)$.

Then the proposed fuzzy modules shown in Figure 3.2 (b)) are inserted into various positions of the baseline model. In Figure 3.2 (a)), when the fuzzy modules are in the location of (1)-(3), the segmentation model is called 'Down_fuzzy' model; when the fuzzy modules are in the location of (4)-(5), the segmentation model is called 'Up_fuzzy' model; when the fuzzy modules are in the location of (1)-(5), the segmentation model is called 'All_fuzzy' model.

For Price's method [6], the basic model architecture was exactly the same. The key difference was the fuzzy module. In their method, feature maps were sorted in descending order by the entropy algorithm and then the top five feature maps were selected to be aggregated via six OWA operators. Their models are also named as 'Down_fuzzy' model, 'Up_fuzzy' model and 'All_fuzzy' model during

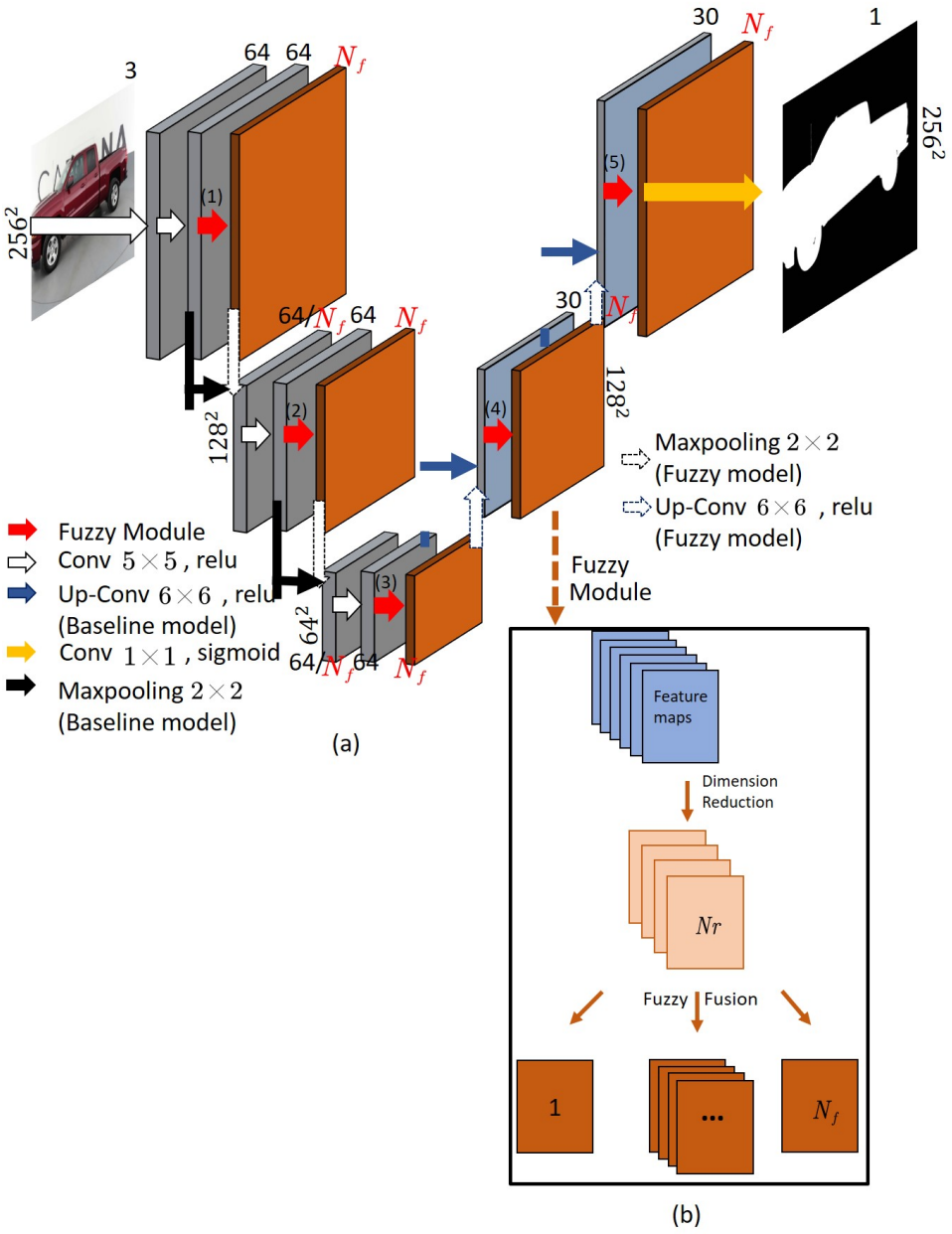


Figure 3.2: (a) The architecture of the CNN model: grey blocks and white arrows mean the down-sampling process, blue blocks and arrows represent up-sampling process, red arrows stand for the concrete location of fuzzy module. When there is no fuzzy module, the CNN model is called baseline model; (b) This part is the details of fuzzy module including dimensionality reduction layers and fuzzy fusion layers. N_r means the number of feature maps after applying the dimensionality reduction operator. N_f represents the number of fuzzy integral operators. Each fuzzy fusion operator generates a new feature map. Black arrows mean the maxpooling operator for the baseline model and the number of output feature maps is 64. Black dashed arrows mean the maxpooling operator for fuzzy models and the number of output feature maps is N_f.

the experiments.

Table 3.1 presents the results of the two comparison experiments: comparing the proposed models with the baseline model and the corresponding models in Price's method [6].

DICE is dice coefficient which is widely used to evaluate the performance of image segmentation. The formula is

$$DICE = \frac{2|A \cap B|}{|A| + |B|} \tag{3.2}$$

where A and B refer to the foreground regions of the predicted image and the ground truth image respectively. Column 'Parameters' in Table 3.1 implies the number of model parameters to be learned, which reflect the model complexity.

By comparing our method with the baseline method, it is seen in Table 3.1 that the 'Up_fuzzy' model consistently achieved better segmentation accuracy than the baseline model with statistical significance (measured by Wilcoxon sign rank test with $p < 0.05$) and a reduction of the model parameter by 4.9%. When using our 'All_fuzzy' model, it was statistically better than the baseline model for the skin dataset and produced better but not statistically significant results than the baseline model for the other two datasets. Remarkably, the 'All-fuzzy' model only used less than 43% parameters of the baseline model.

By comparing the segmentation results of the proposed fuzzy models with those of Price's fuzzy models, the proposed fuzzy modules is considerably better than the Price's methods with statistical significance. Although it shows that the proposed fuzzy modules require more parameters (10%–15%), the segmentation performance 'Down_fuzzy' 'Up_fuzzy' and 'All_fuzzy' models were all improved significantly using our fuzzy modules (7.52% increase, 4.42% increase and 27.29% increase for Skin dataset, 4.98% increase, 3.13% increase

Table 3.1: Experimental results for baseline model, Price's fuzzy models and the proposed fuzzy models. Mean $\pm$ standard deviation values are reported for Dice measure. The number of model parameters are also listed. $\sharp$ means the proposed model and Price's model are significantly different measured by wilcoxon sign rank test with P value < 0.05; $*$ means the proposed model and the baseline model are significantly different with P value < 0.05

Datasets	Baseline model		Fuzzy models	Price's method [6]		the proposed method	
	DICE(%)	Parameters		DICE(%)	Parameters	DICE(%)	Parameters
Skin	80.31 ± 0.31	0.41M	Down_fuzzy	74.42 ± 0.95	0.17M	80.02 ± 0.26 $\sharp$	0.19M
			Up_fuzzy	77.57 ± 0.96	0.39M	81.00 ± 0.29 $\sharp *$	0.39M
			All_fuzzy	64.86 ± 7.92	0.14M	82.56 ± 0.60 $\sharp *$	0.17M
Nuclei	88.91 ± 0.04	0.41M	Down_fuzzy	84.87 ± 0.21	0.17M	89.10 ± 0.49 $\sharp$	0.19M
			Up_fuzzy	87.36 ± 0.68	0.39M	90.09 ± 0.49 $\sharp *$	0.39M
			All_fuzzy	76.13 ± 3.37	0.14M	89.24 ± 0.28 $\sharp$	0.17M
Cars	99.12 ± 0.01	0.41M	Down_fuzzy	90.43 ± 0.08	0.17M	99.06 ± 0.01 $\sharp$	0.19M
			Up_fuzzy	98.86 ± 0.09	0.39M	99.25 ± 0.02 $\sharp *$	0.39M
			All_fuzzy	92.75 ± 0.25	0.14M	99.17 ± 0.03 $\sharp$	0.17M

and 17.22% increase for Nuclei dataset, 9.54% increase, 0.39% increase and 6.92% increase for Cars dataset compared with the Price's fuzzy models).

Figure 3.3 shows some segmentation results for the three datasets. Figure 3.3 (a)(b) are the original image and the corresponding ground truth annotation, respectively; Figure 3.3(c) is the predicted image segmentation of the baseline model; Figure 3.3(d)(f)(h) are the predicted image segmentation based on Price's method [6]; Figure 3.3(e)(g)(i) are the predicted image segmentation based on the proposed method. By comparing Figure 3.3(d)(f)(h) and Figure 3.3(e)(g)(i), the proposed method produced more accurate segmentation boundary than Price's method [6] which are more similar to the ground truth annotation qualitatively.

3.3.3 Application of Fuzzy Integral Module in UNet

In order to verify the generalizability of the proposed fuzzy integral module, a commonly-used semantic image segmentation model—UNet [1] is chosen. UNet used skip connection module to connect the outputs of convolutional layers and de-convolutional layers, which is beneficial for directly transmitting features learning from convolutional layers to de-convolutional layers. Figure 3.4 shows the UNet architecture. It consists of two parts: encode and decode. Double-conv1-5 belong to the encoding process which is designed to extract features. The decoding process including Doule-conv6-9 is devised to enable precise localization and restore image size by transposed convolutions. Herein, as the Double-conv layer is constituted by two convolutional operators, the proposed fuzzy integral module is inserted between these two convolutional operators. Table 3.2 presents the results of the fuzzy module in the encoding and decoding process. 'UNet' refers to original UNet without fuzzy module; 'Up_fuzzy' means only Double-conv6-9 layers include fuzzy integral module;

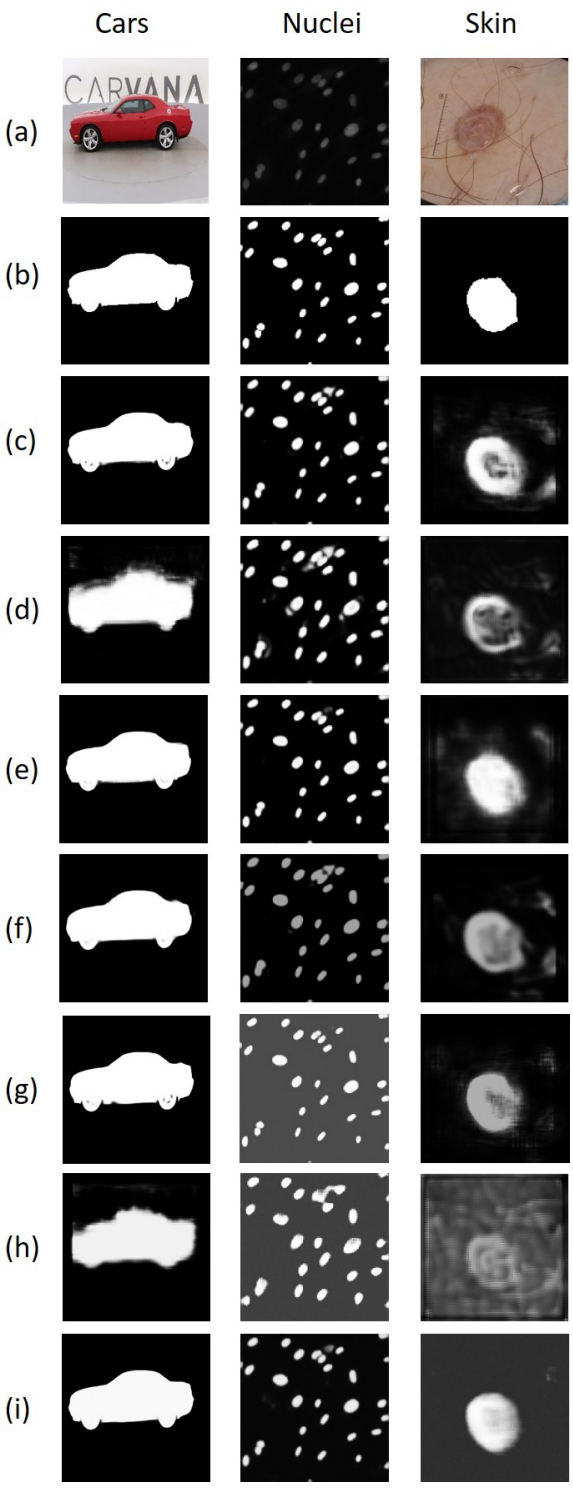


Figure 3.3: Segmentation results: (a) original image; (b) benchmark image; (c) predicted image segmentation of baseline model; (d) predicted image segmentation of 'Down_fuzzy' model based on Price's method [6]; (e) predicted image segmentation of 'Down_fuzzy' model based on the proposed method; (f) predicted image segmentation of 'Up_fuzzy' model based on Price's method [6]; (g) predicted image segmentation of 'Up_fuzzy' model based on the proposed method; (h) predicted image segmentation of 'All_fuzzy' model based on Price's method [6]; (i) predicted image segmentation of 'All_fuzzy' model based on the proposed method.

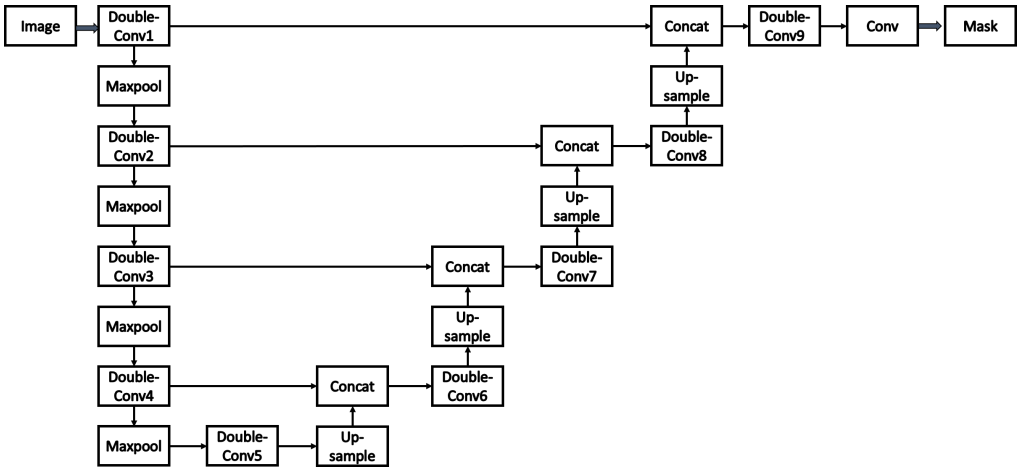


Figure 3.4: The pipeline of UNet

Table 3.2: Experimental results for the widely-used segmentation model UNet with the proposed fuzzy integral module.

Datasets	Model	DICE(%)	Parameters
Skin	UNet	87.21 ± 0.11	31.04M
	Up_fuzzy	87.04 ± 0.19	21.77M
	Down_fuzzy	86.45 ± 0.45	12.35M
	All_fuzzy	85.65 ± 0.24	3.084M
Nuclei	UNet	91.17 ± 0.19	31.04M
	Up_fuzzy	90.97 ± 0.12	21.77M
	Down_fuzzy	90.17 ± 0.51	12.35M
	All_fuzzy	89.47 ± 0.45	3.084M
Cars	UNet	98.88 ± 0.01	31.04M
	Up_fuzzy	98.77 ± 0.03	21.77M
	Down_fuzzy	98.59 ± 0.10	12.35M
	All_fuzzy	98.36 ± 0.01	3.084M

'Down_fuzzy' represents the fuzzy integral module is added in Double-conv1-5 layers; 'All_fuzzy' refers to that all Double-conv layers consist of the fuzzy integral module. DICE is the performance index for image segmentation. Parameters imply the number of parameters for the proposed model, which is given to measure the model complexity.

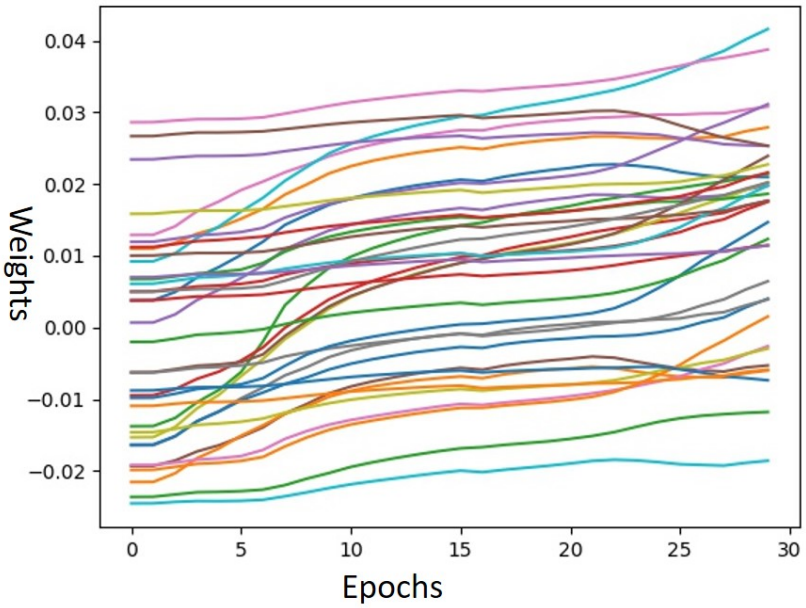
Note that the target of the proposed fuzzy integral module is to integrate the learned features in varied ways rather than extract features from given data. Table 3.2 shows that applying the fuzzy module during the encoding phase results in performance degradation while that during the decoding process causes performance improvement or equal performance by comparing with the baseline model. Therefore, it is more appropriate and reasonable to arrange the fuzzy module in the decoding stage, the function of which is to restore image resolution. However, in this experiment, a fixed fuzzy module with five filters in the dimension reduction layer and six OWA operators in the fuzzy fusion layer is applied. With the increasing amount of filters and fusion operators, information loss will be mitigated to some extent which probably leads to performance enhancement in the stage of encoding.

Furthermore, the last column of Table 3.2 shows that fuzzy modules are capable of reducing 29.86% parameters in the decoding stage, 60.21% parameters in the encoding process, and 90.06% parameters in the whole U-net model, respectively. Moreover, the performance of 'Down_fuzzy' and 'All_fuzzy' models did not fall dramatically (0.87% drop and 1.78% drop for skin dataset, 1.09% drop and 1.86% drop for nuclei dataset, 0.29% drop and 0.52% drop for cars dataset compared with the baseline model), while the number of model parameters saw a significant decrease. It suggests that numerous parameters in the original UNet are redundant and the proposed fuzzy integral module is capable of reducing redundancy.

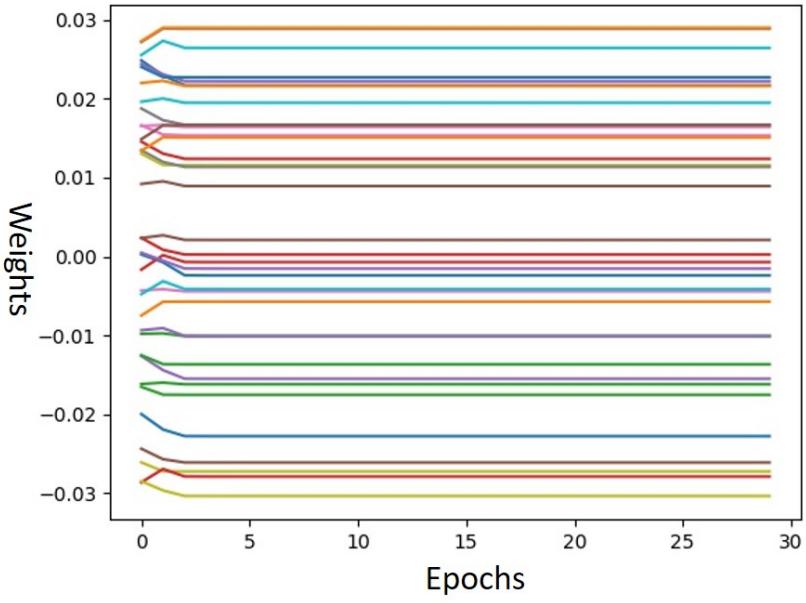
3.4 Discussion

Table 3.1 and 3.2 show that the proposed fuzzy module has the ability to reduce the model complexity. The reason is explained using Figure 3.2. In Figure 3.2(a), one convolutional operator in the baseline model requires $64 \times 5 \times 5 \times 64$ (Bias parameters are neglected in this calculation for simplicity) weights during the encoding stage, while after adding the proposed fuzzy module, the number of weights is reduced to $64 \times 5 \times 5 \times (N_r + N_f)$. When the sum of $(N_r + N_f)$ is smaller than 64, fewer parameters are required by adding the fuzzy module. Therefore, the proposed fuzzy module can be treated as a viable solution if a real-world application requires light-weight models.

Table 3.1 also shows that our fuzzy module is superior to the Price's fuzzy method [6] for 'Down_fuzzy', 'Up_fuzzy', 'All_fuzzy' models. The reason is that Price's method, choosing the top 5 feature maps and discarding other feature maps, undermines the loss propagation process, while the proposed model has the ability to eliminate this drawback. Examples of the weight updating process for the proposed model and Price's model are given in Figure 3.5. It is noted that for simplicity, only 36 weights in one kernel are chosen to be visualized. Other kernels share the same situation. Figure 3.5(a) shows that all of the chosen weights in the proposed new model are updated regularly. Figure 3.5(b) shows that the updating process of the given weights in Price's model is not successively updated: from epoch 1-3, they are updated, and after epoch 3 they remain unchanged. Since when one feature map is discarded, the loss cannot be back propagated successfully that results in the corresponding connected weights remaining the same value. The crucial and valuable characteristic of deep CNN models is that they are capable of learning weights automatically. If a new module that is introduced in the deep CNN models has a negative effect on the weights learning process, the performance will be heavily impacted.



(a) The weight updating process in proposed model



(b) The weight updating process in original model [6]

Figure 3.5: The weights updating process of one filter for the proposed model and the Price's model [6] (a) means that all of the chosen weights in the proposed new model are updated regularly, (b) shows that the chosen weights in the Price's model [6] are not successively updated.

This explains why Price's method performs worse in comparison with the baseline model (shown in Table 3.1). Our new fuzzy module not only helps reduce the quantities of feature maps but also keeps all the weights updated regularly.

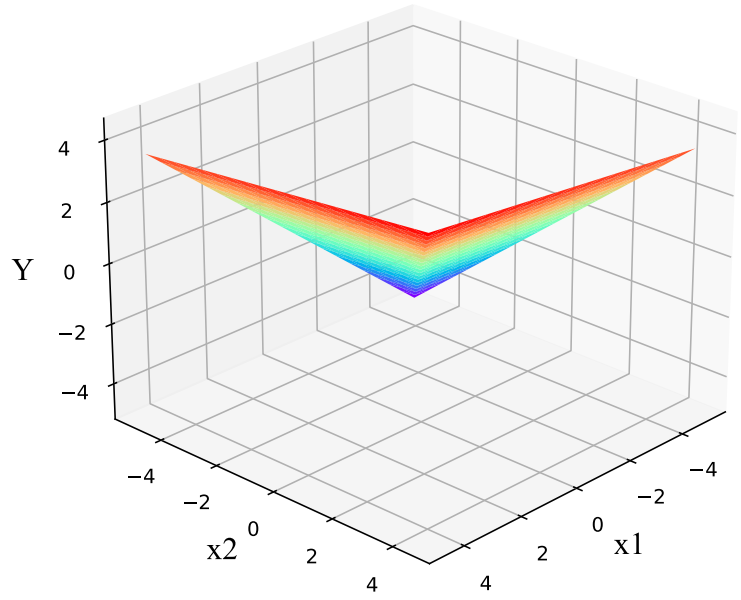
For the second experiment, a fixed fuzzy module with five filters and six OWA operators is inserted into the commonly-used segmentation model–UNet. Better segmentation performance may be obtained using other deeper architectures or more filters and OWA operators. However, the main focus of the second experiment is to demonstrate whether the proposed fuzzy module still works in the commonly-used segmentation model. Hence, the fuzzy module remains consistent with the first experiment.

Moreover, in the proposed fuzzy integral module, a special case of fuzzy integrals namely ordered weight averaging (OWA) to merge information at the feature level. The first step is to sort the feature maps and then use pre-trained weights to fuse the feature maps. In order to avoid the influence of subjective factors, it is worth further investigating whether the pre-trained weights can be learnable.

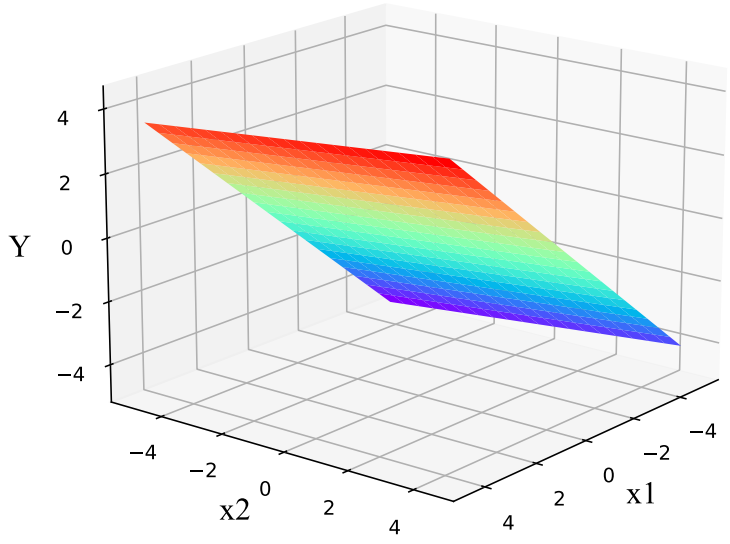
For example, given two feature maps x_1 and x_2 and two weights ω_1 and ω_2, the OWA operator is

$$Y_{\mathrm{OWA}} = \begin{cases} \omega_1 x_1 + \omega_2 x_2, x_1 \geqslant x_2 \\ \omega_1 x_2 + \omega_2 x_1, x_1 < x_2 \end{cases} \tag{3.3}$$

Due to ω_1 and ω_2 are predetermined, the Y_{OWA} is shown in Figure 3.6 (a). If the ω_1 and ω_2 are learnable, the Y_{OWA} in Figure 3.6(a) possibly turns into a plane shown in Figure 3.6 (b), in which the nonlinear characteristic would be disappeared. Moreover, as there is no prior information for the learnable weights, the OWA fusion operator will be uninterpretable. However, the predetermined weights potentially result in decreased accuracy when targeting complicated vision tasks. Herein, only binary class datasets are considered, more complicated



(a) The OWA operators with pre-defined weights



(b) The OWA operators with learnable weights

Figure 3.6: The visualization of the fusion process of two elements

datasets will be applied to verify the efficiency of the proposed fuzzy module in future work.

3.5 Summary

In this chapter, a new fuzzy integral module that can be integrated into semantic segmentation models is proposed to address objective 1. The fuzzy integral module consists of a dimensionality reduction operator and OWA operators. The dimensionality reduction operator is a convolutional layer, which can help reduce the number of feature maps without affecting the process of loss back-propagation. OWA operators are executed to generate new feature maps by fusing along the feature channel dimension. Compared to another fuzzy-integral-based semantic segmentation model (Price's method) [6], the proposed module is more efficient and achieves better segmentation performance. Note that in order to make a fair comparison with Price's method [6], the same semantic segmentation model was adopted in our first experiment. However, this semantic segmentation model used in Price's method [6] is a toy model and not a widely-used image segmentation model. Thus, our second experiment applied the proposed fuzzy integral module to the UNet model to verify its generalizability. From the experimental results, the UNet model complexity is considerably reduced especially when fuzzy integral modules are inserted in the encoding process, while the segmentation performance remains similar.

After the exploration of semantic segmentation models, the collected data will be utilized to train the model. During training time, the loss function plays a significant role in weight updating. Thus, in the next chapter, the fuzzy-based loss function for semantic segmentation is investigated.

Chapter 4

Boundary-wise Loss for Semantic Segmentation Based on Fuzzy Rough Sets

In semantic segmentation, the loss function plays an important role as it determines the segmentation model convergence behavior and performance. According to the literature in Section 2.2.2, many past and current methods utilize pixel-wise (e.g. cross-entropy) and region-wise (e.g. dice) losses while boundary-wise loss is underexplored. It is known that one of the key aims of semantic segmentation is to precisely delineate objects' boundaries. Hence, it is essential to design a loss function that measures the errors around objects' boundaries. Fuzzy rough sets are constituted by the fuzzy equivalence relation, which is commonly used to measure the difference between two sets. Thus, in this chapter, by addressing objective 2, fuzzy rough sets are proposed to construct the boundary-wise loss in semantic segmentation models for the first time, as presented in our study [78].

Section 4.1 introduces the motivation and the importance of the boundary-wise

loss compared to other loss functions. The mathematical derivation of boundary-wise loss based on the lower approximation of fuzzy rough sets in semantic segmentation models is given in Section 4.2. In Section 4.3, experiments with various segmentation models and datasets are conducted, which aims to compare the performance of the proposed fuzzy-rough-sets-based boundary-wise loss function with the other four loss functions. Section 4.4 further investigates the reason why the proposed loss function performs better than other boundary-wise loss functions by the comparison of the loss variation curves in the training process and testing process. The contributions and conclusion of this chapter are summarized in Section 4.5.

4.1 Background and Motivation

As mentioned in Section 2.2.2, the loss function (also called the cost function or error function) is a function that measures the difference between the predicted values and the actual values. For deep learning optimization problems, the segmentation network parameters are estimated by minimizing the given loss function iteratively in a training process. The loss function plays a considerably important role in the training process of semantic segmentation networks as it guides the convergence process and affects the performance of neural networks.

Researchers have designed various types of loss functions to address semantic segmentation problems. Cross entropy loss [19] and dice coefficient loss [20] are the commonly used image segmentation losses. Cross entropy loss is a type of pixel-wise loss, which is calculated by the negative average of the log of corrected predicted probabilities. This loss focuses on the predicted value of each pixel and performs less robustly for unbalanced data. Thus many cross-entropy variation losses are proposed to handle unbalanced data issues, includ-

ing balanced cross-entropy loss [21], focal loss [22]. Dice coefficient loss is a region-wise loss that quantifies the intersection regions of the predicted segmentation and the ground truth segmentation. This loss performs well on unbalanced datasets but its training error curve is unstable and gives no information for the convergence procedure. To take advantage of both dice and cross-entropy loss, Taghanaki et al. [23] introduced a hybrid loss that combines the dice loss and cross-entropy loss.

All the aforementioned losses are pixel-wise and region-wise losses. In image segmentation tasks, uncertainty and misclassification normally happen at the boundaries [26]. Therefore, if a loss function is designed to concentrate on the boundary, the image segmentation performance can be potentially improved. The research conducted by Karimi et al. [43] discussed the boundary difference between the predicted segmentation and the ground truth segmentation. In their study, distance transforms and morphological operations were applied to construct the semantic segmentation boundary-wise loss, which calculated the Hausdorff distance between the predicted image boundary and the ground truth image boundary. However, Karimi et al. [43] only studied the performance of a combined loss based on region-wise and boundary-wise losses without an in-depth investigation of the boundary-wise loss itself.

On the other hand, as mentioned in Section 2.4.2, in 1990, Dubois et al. [24] introduced a novel theory named fuzzy rough sets which combined fuzzy sets with rough sets. The key idea of the fuzzy rough sets is to supersede the equivalence relation of rough sets by a fuzzy equivalence relation [79]. Thus, the fuzzy rough sets have the ability to manage data with fuzziness and vagueness based on the similarity of different attributes. As the generalizations of classical rough sets, fuzzy rough sets are commonly used in image segmentation [80, 81], dimensionality reduction [82], feature selection [83, 84], etc. Although fuzzy rough sets are applied widely in a multitude of AI tasks, no research proposed

the use of fuzzy rough sets as the loss function in machine learning models. In this chapter, as the lower approximation of fuzzy rough sets has the ability to evaluate the difference between two sets, a novel boundary-wise loss is proposed to address the limitations of other methods mentioned above based on the lower approximation of fuzzy rough sets. The proposed novel loss function addresses objective 2 in this thesis and the following is a detailed presentation of the function.

4.2 A New Boundary-wise Loss Function for Semantic Segmentation

In this section, a new boundary-wise loss for semantic segmentation is proposed. Based on the theory of fuzzy rough sets in Section 2.4.2, the lower approximation $\underline{R_s}D_i(x)$ of fuzzy rough sets means the degrees the x certainly belongs to the set D_i. Therefore, the sum $\left(\sum_{x\in X}\underline{R_s}D_i(x)\right)$ can be applied to evaluate the similarity between two sets X and D_i. The key point of the boundary-wise loss is to calculate the difference between the set of predicted image boundary pixels and the set of ground truth image boundary pixels. It is a natural idea to use the lower approximation of fuzzy rough sets to calculate the boundary-wise loss. The detailed mathematical derivation is as follows.

4.2.1 Lower Approximation of Fuzzy Rough Sets

Given a finite and nonempty set of samples U, and decision $\mathbb{D}$ which partitions the samples into subsets $\{D_1, D_2, \cdots, D_M\}$. For $\forall x \in U$, if $x \notin D_i(x)$, $D_i(x) = 0$, otherwise $D_i(x) = 1$. Based on the definition of fuzzy rough sets in Section 2.4.2, the membership degree of x certainty belonging to the

given class D_i is calculated by the lower approximation of fuzzy rough sets $\underline{R_s}D_i(x) = \underset{y\in U}{inf}\, s(1-R(x,y), D_i(y))$, where s refers to the s-norm and $R(x,y)$ is the fuzzy equivalence relation between samples.

To further simplify the equation $\underline{R_s}D_i(x) = \underset{y\in U}{inf}\, s(1-R(x,y), D_i(y))$, s-norms in Table 2.1 are used:

- when s-norm operator is $s_M(a,b) = \max(a,b)$, the lower approximation is calculated by

$$\begin{aligned}
\underline{R_s}D_i(x) &= \underset{y\in U}{inf}\, s_M(1-R(x,y), D_i(y)) \\
&= \underset{y\in U}{inf} \max(1-R(x,y), D_i(y)) \\
&= \underset{y\in U}{inf} \begin{cases} 1, y \in D_i \\ 1-R(x,y), y \notin D_i \end{cases} \\
&= \underset{y\notin D_i}{inf} (1-R(x,y))
\end{aligned} \tag{4.1}$$

- when s-norm operator is $s_P(a,b) = a+b-ab$, the lower approximation is calculated by

$$\begin{aligned}
\underline{R_s}D_i(x) &= \underset{y\in U}{inf}\, s_P(1-R(x,y), D_i(y)) \\
&= \underset{y\in U}{inf} (1-R(x,y)+D_i(y)-(1-R(x,y))\times D_i(y)) \\
&= \underset{y\in U}{inf} \begin{cases} 1, y \in D_i \\ 1-R(x,y), y \notin D_i \end{cases} \\
&= \underset{y\notin D_i}{inf} (1-R(x,y))
\end{aligned} \tag{4.2}$$

- when s-norm operator is $s_L(a,b) = \min(a+b,1)$, the lower approxima-

tion is calculated by

$$\begin{aligned}\underline{R_s}D_i(x) &= \inf_{y\in U} s_L\left(1-R(x,y), D_i(y)\right)\\ &= \inf_{y\in U} \min\left(1-R(x,y)+D_i(y), 1\right)\\ &= \inf_{y\in U}\begin{cases} 1, y\in D_i \\ 1-R(x,y), y\notin D_i \end{cases}\\ &= \inf_{y\notin D_i}\left(1-R(x,y)\right)\end{aligned} \tag{4.3}$$

- when s-norm operator is $s_{\cos}(a,b) = \min\left(a+b-ab+\sqrt{2a-a^2}\sqrt{2b-b^2}, 1\right)$, the lower approximation is calculated by

$$\begin{aligned}\underline{R_s}D_i(x) &= \inf_{y\in U} s_{\cos}\left(1-R(x,y), D_i(y)\right)\\ &= \inf_{y\in U} \min(1-R(x,y)+D_i(y)-(1-R(x,y)\\ &\times D_i(y)+\sqrt{1-R^2(x,y)}\sqrt{2D_i(y)-D_i^2(y)}, 1)\\ &= \inf_{y\in U}\begin{cases} 1, y\in D_i \\ 1-R(x,y), y\notin D_i \end{cases}\\ &= \inf_{y\notin D_i}\left(1-R(x,y)\right)\end{aligned} \tag{4.4}$$

From the equation 4.1—4.4, the lower approximation of fuzzy rough sets has the consistent representation $\underline{R_s}D_i(x) = \inf_{y\notin D_i}\left(1-R(x,y)\right)$ regardless of the s-norm operator. $\underline{R_s}D_i(x)$ means that the degree of x certainty belonging to D_i relies on the closest sample of another category. Consider a special situation in which there are only two classes D_1 and D_2. The formula $\underline{R_s}D_1(x) = \inf_{y\in D_2}\left(1-R(x,y)\right)$ denotes the likelihood of sample x belonging to class D_1 increases with the distance between x and class D_2. It indicates if x is far apart from class D_2, then it is more likely to belong to class D_1.

4.2.2 Fuzzy Rough Sets Loss

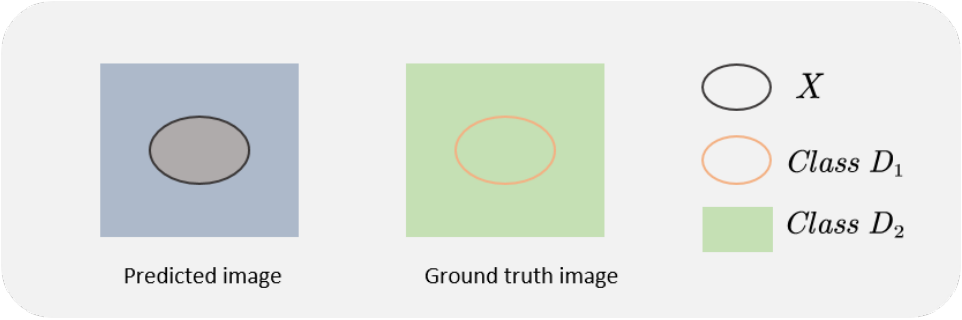


Figure 4.1: The left part refers to the predicted image and the right part is the ground truth image. Black circle is the predicted image boundary pixel set; orange circle is the ground truth image boundary pixel set and belongs to the class D_1; the rest parts of the ground truth image pixels belong to the class D_2; grey part is the area of segmentation object.

For a semantic segmentation model, the training process is that given the input image array $X \in \Re^{h\times w\times c}$ (h, w and c refer to the height, weight and channel of X), the output predicted image is obtained by

$$\hat{Y} = f(W \otimes X), \tag{4.5}$$

where $\hat{Y} \in \Re^{h'\times w'\times \mathcal{P}}$ (h' and w' are the height and weight of $\hat{Y}$, $\mathcal{P}$ is the number of pixel categories), W represents the model parameters, and f means the soft-max function. Then the loss value is back-propagated to update the parameters. The abstract loss function is written as $loss = F\left(\hat{Y}, Y\right)$ where $\hat{Y}$ and Y are the predicted image and the corresponding ground truth image, and F means the selected loss function. The loss value increases with the distance between $\hat{Y}$ and Y.

Figure 4.1 shows the predicted image and the corresponding ground truth image. For the ground truth image, the boundary pixel sets belong to class D_1, while the rest parts of the pixels are assigned to class D_2. In the predicted image, the boundary pixel sets are $\{x_i : x_i \in X\}$. For the binary categories semantic segmentation, the ground truth image only has two categories D_1 and D_2 ($D_1 \cap$

$D_2 = \oslash$, $D_1 \cup D_2 = \mho$, where $\mho$ refers to the whole ground truth image) so that the similarity between X and D_2 can be applied to assess the distance of X and D_1. The greater the similarity value of X and D_2 is, the farther X is from D_1, which means the loss difference between X and D_1 is greater. Therefore, the boundary loss is represented as

$$loss_P = \mathcal{L}\left(\hat{Y}_X, Y_{D_2}\right). \tag{4.6}$$

where $\mathcal{L}$ is the similarity of X and D_2.

From Section 4.2.1, the lower approximation $\underline{R_s}D_i(x) = \underset{y \notin D_i}{inf}\left(1 - R(x,y)\right)$ of fuzzy rough sets means the degrees the x certainly belongs to the set D_i. Therefore, given the $\{x_i : x_i \in X\}$ and $D_i = D_2$, the similarity of X and D_2 is assessed by the average of the lower approximations for all x: $\frac{1}{m}\left(\sum_{x \in X} \underline{R_s}D_2(x)\right)$ where m is the number of X. The boundary loss is evaluated by

$$\begin{aligned} loss_P &= \frac{1}{m}\left(\sum_{i=1}^{m} \underline{R_s}D_2(x_i)\right) \\ &= \frac{1}{m}\sum_{i=1}^{m}\left(\begin{array}{c} inf \\ j \in \{1,2,...,k\} \\ y_j \notin D_2 \end{array} \left(1 - R\left(x_i, y_j\right)\right)\right) \end{aligned} \tag{4.7}$$

where $x_i \in X$, $y_j \notin D_2$, R is the fuzzy equivalence relation of x_i, y_j. Furthermore, based on Figure 4.1, the ground truth image only has two categories D_1, D_2 and $D_1 \cap D_2 = \oslash$, so $y_j \notin D_2 \Leftrightarrow y_j \in D_1$. It means that the boundary loss can be calculated by the boundary sets of the predicted image and the ground truth image. The widely used Gaussian kernel $R(x,y) = \exp\left(-\frac{\|x-y\|^2}{\sigma}\right)$ [60] is applied to calculate the fuzzy equivalence relation. The final boundary-wise

loss formula is obtained by

$$loss_P = \frac{1}{m}\sum_{i=1}^{m}\left(\begin{array}{c} inf \\ j\in\{1,2,...,k\} \\ y_j \in D_1 \end{array} \left(1-R\left(x_i,y_j\right)\right)\right)$$
$$= \frac{1}{m}\sum_{i=1}^{m}\left(\begin{array}{c} inf \\ j\in\{1,2,...,k\} \\ y_i \in D_1 \end{array} \left(1-\exp\left(-\frac{\|x_i-y_j\|^2}{\sigma}\right)\right)\right) \tag{4.8}$$

and named as Fuzzy Rough Sets Loss (FRSLoss).

To make the proposed loss more robust and satisfy the symmetric condition, the predicted image is separated into two categories $\hat{D}_1$, $\hat{D}_2$ and the boundary pixels sets of the ground truth image are the samples to be classified. Then the average of the lower approximation for all boundary pixels of the ground truth is adopted to calculate the boundary-wise loss:

$$loss_G = \frac{1}{k}\left(\sum_{j=1}^{k}\underline{R_s}\hat{D}_2\left(y_j\right)\right)$$
$$= \frac{1}{k}\sum_{j=1}^{k}\left(\begin{array}{c} inf \\ i\in\{1,2,...,m\} \\ x_i \in \hat{D}_1 \end{array} \left(1-R\left(y_j,x_i\right)\right)\right) \tag{4.9}$$

It is noted that $Loss_P \neq Loss_G$. Therefore, the Dual Fuzzy Sets Loss (DFRSLoss) is introduced as $Loss_D = (Loss_P + Loss_G)$.

The aforementioned loss is only suitable for binary classification: one category is the object pixels and the other is the background pixels. Nevertheless, numerous practical segmentation tasks are required to address multi-class issues. Therefore, to expand the application scope of the proposed loss, the binary formula is extended to calculate the multi-class segmentation loss. Given the number of the categories is $\mathcal{P}$, the multi-class segmentation can be divided into $\mathcal{P}$ binary-class segmentation tasks. Thus, the multi-class FRSLoss is:

$$\begin{aligned} Loss_M &= \frac{1}{\mathcal{P}} \sum_{\rho=1}^{\mathcal{P}} \frac{1}{m} \left(\sum_{i=1}^{m} \underline{R_s} D_2^{\rho} \left(x_i^{\rho} \right) \right) \\ &= \frac{1}{\mathcal{P}} \sum_{\rho=1}^{\mathcal{P}} \frac{1}{m} \left(\sum_{i=1}^{m} \left(\begin{array}{c} inf \\ j \in \{1,2,\cdots,k\} \\ y_j \in D_1^{\rho} \end{array} \left(1 - R\left(x_i^{\rho}, y_j^{\rho} \right) \right) \right) \right) \end{aligned} \tag{4.10}$$

4.2.3 Distance Transform Algorithm

Based on equation (4.8), the FRSloss is one non-convex function. In order to make the loss applicable in the segmentation model, the distance transform algorithm is utilized to calculate the FRSLoss in practical applications.

Given a binary image $\mathfrak{B}$, the aim of the distance transform algorithm is to obtain the closest distance from pixels in the segmented object to the object boundary. After using the distance transform algorithm on $\mathfrak{B}$, a distance map $\mathfrak{D}_f = f_{DT}(\mathfrak{B})$ is generated, where $\mathfrak{D}_f$ and $\mathfrak{B}$ have the same size and the algorithm f_{DT} used to calculate the pixel value in $\mathfrak{D}_f$ consists of three steps:

1. Divide the pixels in the binary image $\mathfrak{B}$ into two sets: the foreground or segmentation object $\Theta = \{(x,y) | \mathfrak{B}(x,y) = 1\}$, and the background

$\Theta^c = \{(x,y)\,|\mathfrak{B}\,(x,y) = 0\}$. (x,y) represents the position of pixels.

2. For the foreground, calculate the pixel value $\mathfrak{D}_f(x,y)$ where $(x,y) \in \Theta$ based on the minimal Euclidean distance from this pixel to Θ^c according to equation (4.11).

$$\begin{aligned}&\mathfrak{D}_f\,(x,y)\\&= \min\left\{\sqrt{(x-x_\alpha)^2+(y-y_\alpha)^2}\,|(x_\alpha,y_\alpha) \in \Theta^c\right\}\end{aligned} \tag{4.11}$$

3. For the background, fill 0 for all pixels belonging to $\{(x,y)|(x,y) \in \Theta^c\}$ in the distance map $\mathfrak{D}_f$.

As the segmentation boundary consists of the closest background pixels around the segmented object, the distance map $\mathfrak{D}_f$ represents the smallest distance from foreground pixels to the segmentation boundary. Moreover, in order to calculate the smallest distance from the background pixels to the segmentation boundary, the distance transform algorithm is also applied to the complementary map $\mathfrak{C} = 1-\mathfrak{B}$ to obtain the corresponding new distance map $\mathfrak{D}_b$. Thus, the final distance map is calculated by $\mathfrak{D} = \mathfrak{D}_f + \mathfrak{D}_b$, which specifies the distance from each pixel to the nearest boundary pixel. Note that for simplicity we ignore the difference of the segmentation boundary during the calculation process of $\mathfrak{D}_f$ and $\mathfrak{D}_b$. Due to the practicability and effectiveness of distance transform in computer vision [85] [86] [87] [88], many researchers devote themselves to finding the optimal algorithm to calculate the distance map. In this chapter, a linear-time algorithm based on min-convolutions and squared Euclidean distance [89] is adopted to calculate the $\mathfrak{D}_f$ and $\mathfrak{D}_b$.

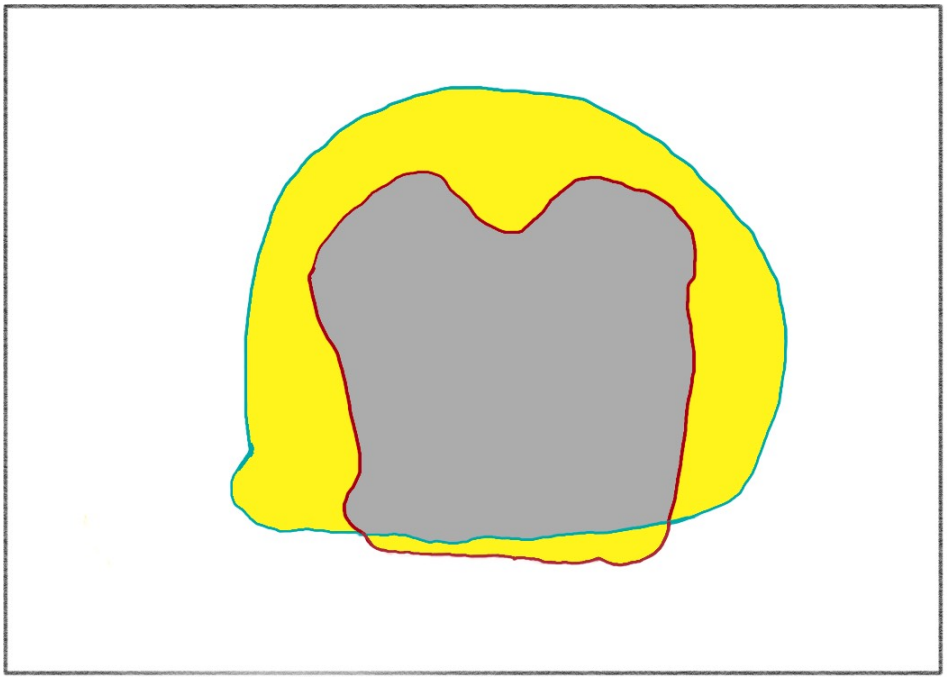


Figure 4.2: The blue line is the object boundary of the ground truth image; the red line is the object boundary of the predicted segmentation image; the grey region is the overlap of the ground truth image and the predicted segmentation image; the yellow region is disjoint parts of the ground truth image and the predicted segmentation image.

4.2.4 Computational Details of Fuzzy Rough Sets Loss

In Figure 4.2, the blue line D_1 is the object boundary of the ground truth image Y; the red line X is the object boundary of the predicted segmentation image $\hat{Y}$; the grey region is the overlap of Y and $\hat{Y}$; the yellow region is disjoint parts of Y

and $\hat{Y}$. From equation (4.8), it can be rewritten as:

$$loss_P = \frac{1}{m}\sum_{i=1}^{m}\left(\begin{array}{c} \underset{j\in\{1,2,...,k\}}{inf} \left(1-\exp\left(-\frac{\|x_i-y_j\|^2}{\sigma}\right)\right) \\ y_i \in D_1 \end{array}\right) = \left(1-\frac{1}{m}\sum_{i=1}^{m}\exp\left(-\frac{\underset{j\in\{1,2,\cdots,k\}}{inf}\|x_i-y_j\|^2}{\sigma}\right)\right) \tag{4.12}$$

where $x_i \in X$, $y_i \in D_1$, m is the number of boundary pixels in predicted image. Thus, the coordinates of all points in X and D_1 should be collected, the process of which is extremely time-consuming, especially in a 256×256 image. In practical application, in order to improve computational efficiency, the closest distance between each pixel x_i in X and D_1:

$$\underset{\substack{j\in\{1,2,\cdots,k\} \\ y_i \in D_1}}{inf} \|x_i-y_j\| \tag{4.13}$$

can be alternatively calculated by the matrix operation

$$|\hat{Y}-Y| \otimes \mathfrak{D} \tag{4.14}$$

where $|\hat{Y}-Y|$ is the yellow region in Figure 4.2 representing disjoint parts of $\hat{Y}$ and Y, $\mathfrak{D}$ is the distance map of the ground truth image and refers to the closest distance from each pixel to the nearest segmentation boundary, $\otimes$ means

pixel-wise multiplication. Equation (4.14) is more efficient than equation (4.13) due to the fact that equation (4.14) does not have to find the predicted image boundary point sets and matrix operation is more applicable and run faster in deep learning models compared to the single-point operation which normally requires numerous loops. Note that the key point of semantic segmentation is to minimize the yellow region in Figure 4.2 and get the red line close to the blue line shown in Figure 4.2. Equation 4.14 represents the closest distance from the pixels in the yellow region to the object boundary. Equation 4.13 refers to the closest distance from the pixels in the red line to the object boundary. Therefore, it is reasonable to replace Equation 4.13 using Equation 4.14.

Furthermore, the loss function should be differentiable. Thus, the distance map $\mathfrak{D}$ is calculated previously and treated as a constant matrix. The absolute value $|\hat{Y} - Y|$ is substituted by $\left(\hat{Y} - Y\right)^2$, and the final loss function formula is written by

$$loss_P = \left(1 - \frac{1}{M}\sum \exp\left(-\frac{\left(\left(\hat{Y} - Y\right)^2 \otimes \mathfrak{D}\right)^2}{\sigma}\right)\right), \tag{4.15}$$

where M is the number of pixels in the yellow region shown in Figure 4.2.

Based on the FRSLoss formula (4.15), the computational procedure of the proposed FRSLoss is described in Algorithm 4.1.

In this way, the non-convex issue of the FRSLoss is handled successfully. Next section, several experiments are conducted to evaluate the effectiveness of FRSLoss.

4.3 Evaluation and Results

In this section, two public datasets with various object shapes and sizes and three widely-used semantic segmentation models including UNet, FCN, and SegNet

Algorithm 4.1 Fuzzy Rough Sets Loss

Input: semantic segmentation model M, raw input image I, corresponding ground truth image Y, the sizes for I and Y are $N \times N$

Output: the loss value FRSLoss

1: input image I into the model M and generate the predicted image P
2: obtain the predicted binary image $\hat{Y}$ using the binarization operator $\hat{Y} = \begin{cases} 1, p \geq 0.5 \\ 0, p < 0.5 \end{cases}$, where p is the pixel value of P
3: calculate the difference map $F = |\hat{Y} - Y|$
4: obtain the complementary map $\overline{Y}$ of the ground truth image Y by using equation $\overline{Y} = 1 - Y$
5: apply the distance transform algorithm on Y and $\overline{Y}$ to get the final distance map $\mathfrak{D}$
6: calculate the minimum distance between predicted image boundary and ground truth image boundary $\Lambda = F \otimes \mathfrak{D}$, where $\otimes$ means pixel-wise multiplication
7: apply Gaussian kernel $k_G = \exp\left(-\frac{|\Lambda|^2}{\sigma}\right)$
8: $FRSLoss =$ **Average**$(1 - \text{Gaussian}(\Lambda))$.
9: **return** FRSLoss

are utilized to assess the performance of the proposed boundary-wise loss. Evaluation metrics, implementation details, and experimental results are described in the following subsections.

4.3.1 Datasets and Metrics

In the next experiments, nuclei and kidney cell datasets were adopted. Both of the datasets have multiple segments in each image. However, the nuclei dataset has more complicated boundaries than the cell datasets (seen in Figure 4.6). The primary merit of the proposed boundary-wise loss is that it pays more attention to the boundary compared with other types of losses. Therefore, to verify the effectiveness and applicability of the proposed new boundary-based loss, it is reasonable and meaningful to choose datasets with diverse boundaries.

- Nuclei dataset comes from the 2018 Data Science Bowl, the goal of which is to segment nuclei boundary from given divergent images (https:

//www.kaggle.com/c/data-science-bowl-2018/data). It includes 670 raw images and the corresponding ground truth images are also available. As the nuclei dataset has multiple segments in each image, its boundaries are relatively complicated and diverse.

- The kidney renal clear cell dataset [90] contains 462 raw images with corresponding ground truth images annotated by experts. The pixel size of each image is 400×400. For sake of avoiding the overfitting issue, data augmentation including flipping and rotation was utilized to increase the quantity of this dataset. Compared with the nuclei dataset, this cell dataset has less complicated boundaries.

To evaluate the segmentation performance of FRSLoss and compare it against other state-of-the-art losses (CELoss, DLoss, HDLoss, and DHDLoss), Dice coefficient (DC) and pixel accuracy (PA) are reported.

$$DC = \frac{2|P \cap R|}{|P| + |R|} \tag{4.16}$$

where P and R present the foreground areas of the ground truth image and the predicted image respectively.

$$PA = \frac{TP + FP}{TP + FP + TN + FN} \tag{4.17}$$

where TN, FN, FP and TP refer to the true negative, false negative, false positive and true positive rates respectively. Note that DC belongs to the region-wise metric and PA is the pixel-wise metric [13].

As the proposed loss, FRSLoss, is a boundary-wise loss, it pays more attention to the boundaries compared to region-wise and pixel-wise losses in theory. Hence, two boundary-wise metrics named 95th-percentile of Hausdorff distance (HD) $d(X,Y) = \max_{x \in X} \min_{y \in Y} \|x - y\|_2$ and average symmetric sur-

face distance (ASD) are also applied to assess the segmentation performance. The formula of ASD is given as:

$$ASD = \frac{1}{\left|S\left(\hat{Y}\right)\right| + |S(Y)|}\Big(\sum_{\alpha \in S(Y)} \min_{\beta \in S\left(\hat{Y}\right)} \|\alpha - \beta\| + \sum_{\beta \in S\left(\hat{Y}\right)} \min_{\alpha \in S(Y)} \|\beta - \alpha\|\Big) \tag{4.18}$$

where $S(Y)$ refers to the boundary pixels set of the ground truth image and $S(\hat{Y})$ is the boundary pixels set of the predicted image, α and β are the elements of boundary sets $S(Y)$ and $S(\hat{Y})$ respectively.

4.3.2 Experimental Design

In this chapter, without loss of generality, three commonly used semantic segmentation models, FCN [13], UNet [1] and SegNet [2], were applied as the backbone and evaluated on two public datasets. FCN is the most classical semantic segmentation model and firstly proposed the encoder-decoder framework for semantic segmentation. In the encoding stage, the semantic segmentation model constituted by convolutional layers, maxpooling layers and relu activation functions is applied to capture the deep features. In the decoding stage, deconvolutional operators are adopted to resize the feature maps in order to make the output image and the ground truth image have the same size. In the FCN model, the size of convolutional filter is 3×3 with the corresponding stride is 1, while the size of deconvolutional filter is 3×3 with the corresponding stride is 2. UNet and SegNet models are the improvement of FCN. They share the same encoder-decoder framework. The difference is that UNet uses skip connection to transmit the features extracted from the encoding stage to the decoding pro-

cess. Whereas, the SegNet introduces an index function to record the maximum value in each sliding window of pooling layers and during the decoding stage uses the index function to recover the feature maps by up-maxpooling operators instead of the deconvolutional operator. In the UNet model, the size of convolutional filter is 3×3 with the corresponding stride is 1, while the size of deconvolutional filter is 2×2 with the corresponding stride is 2. In the SegNet model, the size of convolutional filter is 3×3 with the corresponding stride is 1, while the size of up-maxpooling filter is 2×2 with the corresponding stride is 2.

Numerous semantic segmentation losses have achieved exceptional performance in various sorts of datasets [19, 20]. To verify the effectiveness of the proposed loss, comparison experiments were conducted. Herein, we choose pixel-wise loss (CELoss), region-wise loss (DLoss), and boundary-wise loss (HDLoss and DHDLoss) for comparison. CELoss, DLoss, HDLoss and DHDLoss are calculated by equations (2.1)—(2.4).

The detailed algorithm procedure of the proposed FRSLoss is given in Section 4.2. Moreover, based on equation (4.8), the Gaussian kernel in the FRSLoss has a hyperparameter σ. To determine an optimal value for σ, cross-validation experiments with the different values of $\sigma = \{0.5, 1, 2, 3, 5, 10\}$ were conducted. Figure 4.3 shows variation curves of the performance evaluation indices with various values of σ. The experimental results show that when σ was equal to 1, the semantic segmentation model performed the best. Therefore, in all following experiments, the value of σ was set to 1, which enabled the segmentation models to achieve the best performance.

All the experiments were implemented based on the Pytorch. Adam optimization algorithm was applied to update the weights with an initial learning rate of 0.0001.

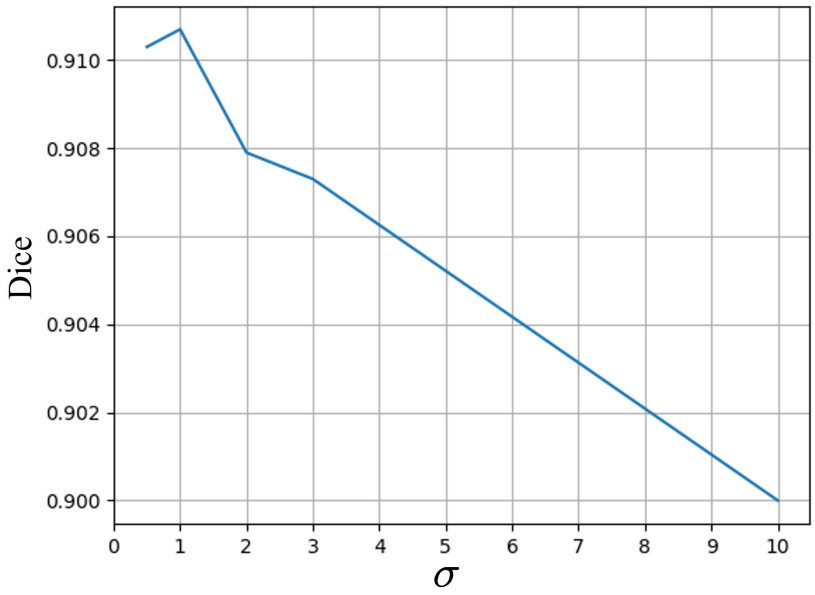


Figure 4.3: Segmentation performance with different σ, when $\sigma = 1$, the semantic segmentation model achieves the best performance.

4.3.3 Results for Nuclei Dataset

Table 4.1 depicts the experimental results for boundary-wise losses in different semantic segmentation models. Dice coefficient (DC) is generally used to assess the segmentation performance and belongs to the region-wise metric, pixel accuracy (PA) is one type of pixel-wise evaluation method, symmetric surface distance (ASD) and 95th-percentile of Hausdorff distance (HD) are adopted to evaluate the boundary distance of the predicted image and the ground truth image. It should be noted that ASD and 95th-percentile of HD have the inverse trend compared to DC and PA: when ASD and 95th-percentile of HD have lower values, segmentation performance is better; while when DC and PA have lower values, segmentation performance is worse.

FRSLoss, and DFRSLoss are the proposed boundary-wise losses. HDLoss and DHDLoss are the other boundary-wise losses proposed by Karimi et al. [43]. To make a fair comparison, FRSLoss and HDLoss only take the distance trans-

Table 4.1: Experimental results for boundary-wise losses, pixel-wise loss and region-wise loss in UNet, FCN and SegNet on the nuclei dataset. Mean ± standard deviation values are reported for all the evaluation measures. $*$ represents FRLoss and CELoss are significantly different with $p < 0.05$; $\sharp$ represents FRLoss and DLoss are significantly different with $p < 0.05$

Model	Loss	DC (%) ↑	PA (%) ↑	ASD (*mm*) ↓	95th-percentile of HD (*mm*) ↓
UNet	FRSLoss	91.02 ± 0.04	$\mathbf{97.64 \pm 0.03}$	$\mathbf{0.82 \pm 0.02}$ $*$	$\mathbf{4.74 \pm 0.20}$ $*\sharp$
	HDLoss	74.75 ± 4.05	94.90 ± 0.41	4.43 ± 0.94	35.78 ± 5.70
	DFRSLoss	$\mathbf{91.07 \pm 0.04}$	97.60 ± 0.02	0.83 ± 0.02	5.09 ± 0.31
	DHDLoss	75.39 ± 2.17	94.49 ± 0.34	2.88 ± 0.22	17.84 ± 1.62
	CELoss	90.97 ± 0.07	97.58 ± 0.01	0.92 ± 0.01	7.83 ± 0.13
	DLoss	90.73 ± 0.13	97.51 ± 0.08	0.83 ± 0.04	5.92 ± 1.01
FCN	FRSLoss	90.39 ± 0.11	97.58 ± 0.04	$\mathbf{0.81 \pm 0.01}$ $\sharp$	$\mathbf{4.10 \pm 0.03}$ $*\sharp$
	HDLoss	82.77 ± 0.60	96.26 ± 0.17	1.81 ± 0.38	8.95 ± 1.73
	DFRSLoss	90.57 ± 0.08	97.62 ± 0.06	0.84 ± 0.01	4.15 ± 0.12
	DHDLoss	83.41 ± 0.45	96.37 ± 0.05	1.35 ± 0.05	5.26 ± 0.28
	CELoss	90.49 ± 0.07	97.55 ± 0.03	0.84 ± 0.01	4.97 ± 0.08
	DLoss	$\mathbf{90.74 \pm 0.10}$	$\mathbf{97.63 \pm 0.02}$	0.86 ± 0.02	5.62 ± 1.35
SegNet	FRSLoss	89.56 ± 0.53	$\mathbf{97.22 \pm 0.12}$	$\mathbf{0.98 \pm 0.06}$ $*\sharp$	6.23 ± 1.28 $*\sharp$
	HDLoss	77.66 ± 2.76	95.11 ± 0.72	3.07 ± 0.59	22.15 ± 2.94
	DFRSLoss	$\mathbf{89.98 \pm 0.15}$	97.19 ± 0.13	0.98 ± 0.12	$\mathbf{5.81 \pm 2.18}$
	DHDLoss	81.93 ± 1.29	95.75 ± 0.31	2.16 ± 0.50	11.18 ± 2.17
	CELoss	89.46 ± 0.26	97.17 ± 0.04	1.14 ± 0.20	8.89 ± 1.74
	DLoss	88.97 ± 0.47	96.98 ± 0.16	1.11 ± 0.08	7.86 ± 0.51

form of the ground truth image into account, while DFRSLoss and DHDLoss consider both directions using the distance maps of the predicted image and the ground truth image. It can be seen in Table 4.1 that the performance of UNet, FCN, and SegNet with FRSLoss and DRSLoss are improved in DC, PA, ASD, and 95th-percentile of HD compared with HDLoss and DHDLoss, which means that the proposed losses are superior to the Hausdroff distance based boundary-wise losses in all given segmentation performance evaluation. Moreover, in comparison of the experimental results of UNet, FCN and SegNet with FRSLoss, HDLoss, DFRSLoss and DHDLoss, the performance variation of the proposed loss function is much smaller across different segmentation models than Hausdroff distance based losses. In addition, the standard deviations of the experimental results indicate that FRSLoss and DFRSLoss are more stable than HDLoss and DHDLoss.

Table 4.1 also shows the experimental results for pixel-wise loss and region-wise loss in different semantic segmentation models. Herein, only widely used pixel-wise (CELoss) and region-wise (DLoss) losses are discussed. As reported in Table 4.1, CELoss and DLoss have similar segmentation accuracy to FRSLoss in DC and PA but FRSLoss performs better in boundary-wise metrics (measured by ASD and 95th-percentile of HD). Figure 4.4 uses boxplots to compare the ASD and 95th-percentile of HD of the proposed boundary-wise loss (FRSLoss), pixel-wise loss (CELoss), and region-wise loss (DLoss). From the boxplots, FRSLoss had a closer boundary distance than CLoss and DLoss. This means that the proposed novel boundary-wise loss method pays more attention to the boundaries than the region-wise and pixel-wise losses. Furthermore, to further verify if there is a statistically significant difference between FRSLoss, CELoss, and DLoss, the Wilcoxon sign rank test with $p < 0.05$ is adopted. In Table 4.1, $*$ represents FRLoss and CELoss are significantly different; $\sharp$ represents FRLoss and DLoss are significantly different. Hence, for DC and PA, no statistical difference between the proposed method and DLoss/CELoss. However, FRLoss has a statistically significant difference with CELoss and DLoss in 95th-percentile of HD for UNet, FCN, and SegNet models, while for the ASD boundary metric, FRSLoss has a statistically significant difference with DLoss on the FCN and SegNet models, with CELoss on the UNet and SegNet models. On the other hand, for different segmentation models, the proposed boundary-wise loss, the widely-used pixel-wise loss, and the region-wise loss have consistent performance: they perform the best in the UNet model and the worst in the SegNet model.

Table 5.2 shows the training time to achieve model convergence of the UNet, FCN, and SegNet models using FRSLoss, DFRSloss, HDLoss, and DHDloss as the loss function for the nuclei dataset. The segmentation models with FRSloss and DFRSloss require less time to reach the optimal state than that with HDLoss

and DHDloss. It means that the proposed boundary-wise losses are capable of boosting the convergence speed compared with other boundary-wise losses.

Table 4.2: Convergence time of the UNet, FCN and SegNet models with different boundary-wise losses

Model	FRSLoss	HDLoss	DFRSLoss	DHDLoss
UNet	**40.71min**	67.17min	**65.10min**	90.43min
FCN	**57.20min**	93.53min	**91.18min**	101.55min
SegNet	**68.57min**	115.12min	**87.45min**	137.95min

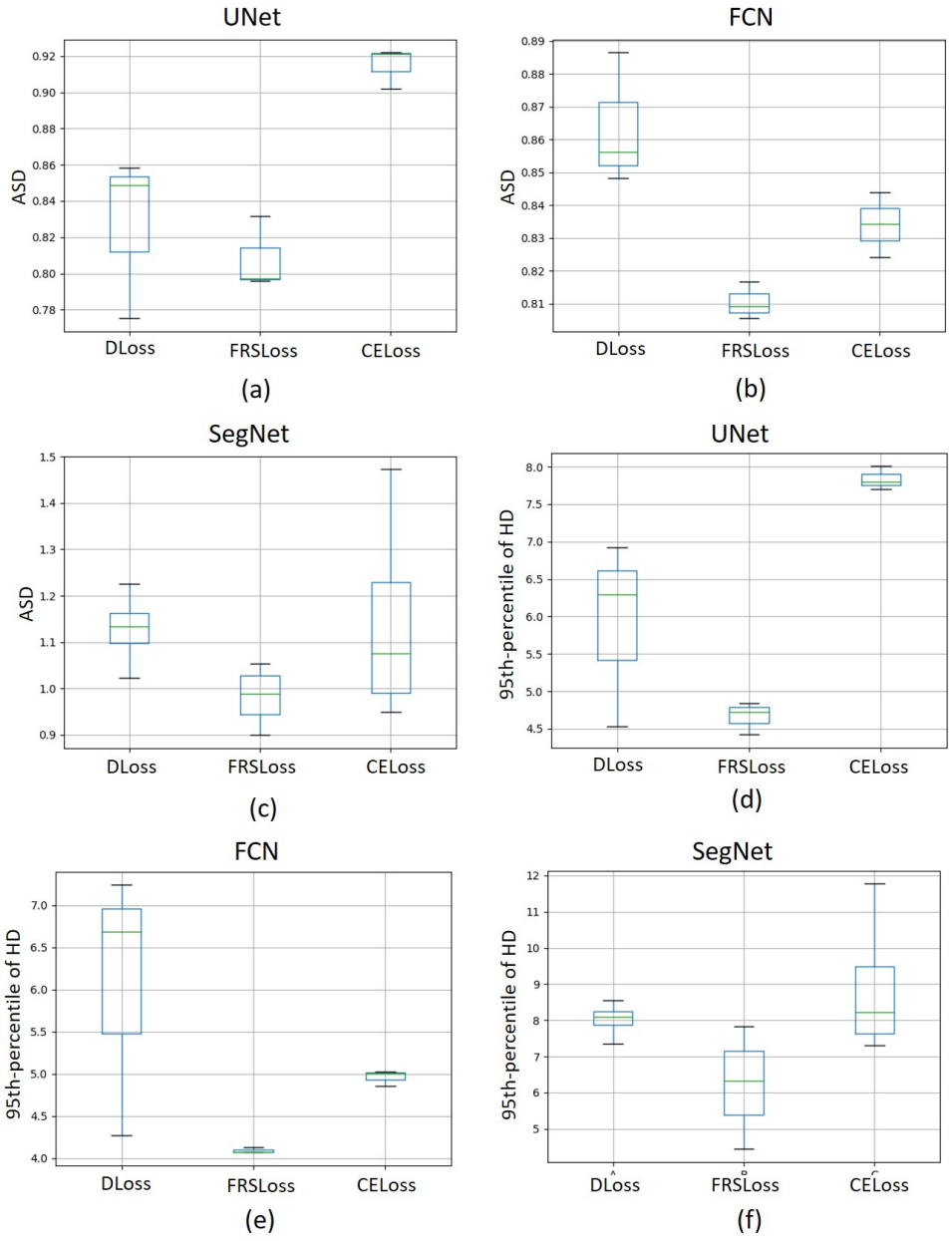


Figure 4.4: Boundary distance metrics ASD and 95th-percentile of HD for DLoss, FRSLoss and CELoss on Nuclei dataset. When the values of ASD and 95th-percentile of HD are lower, the predicted image and the ground truth image would have closer boundary distance.

4.3.4 Results for Cell Dataset

Table 4.3: Experimental results for boundary-wise losses, pixel-wise loss and region-wise loss in UNet, FCN and SegNet on the cell dataset. Mean ± standard deviation values are reported for all the evaluation measures. $*$ represents FRLoss and CELoss are significantly different with $p < 0.05$; ♯ represents FRLoss and DLoss are significantly different with $p < 0.05$

Model	Loss	DC (%) ↑	PA (%) ↑	ASD (*mm*) ↓	95th-percentile of HD (*mm*) ↓
UNet	FRSLoss	75.11 ± 0.21	$\mathbf{92.57 \pm 0.08}$	2.42 ± 0.03 ♯	$\mathbf{16.59 \pm 0.17}$ $*$♯
	HDLoss	72.84 ± 0.13	89.97 ± 0.58	2.93 ± 0.08	21.06 ± 1.46
	DFRSLoss	75.16 ± 0.15	92.56 ± 0.19	$\mathbf{2.42 \pm 0.01}$	16.90 ± 0.09
	DHDLoss	72.79 ± 0.21	91.29 ± 0.98	2.79 ± 0.04	18.34 ± 0.88
	CELoss	75.55 ± 0.12	92.31 ± 0.15	2.43 ± 0.02	17.20 ± 0.09
	DLoss	$\mathbf{75.68 \pm 0.05}$	92.05 ± 0.04	2.47 ± 0.01	18.19 ± 0.22
FCN	FRSLoss	74.65 ± 0.13	$\mathbf{92.64 \pm 0.05}$	2.42 ± 0.00 $*$	$\mathbf{16.15 \pm 0.11}$ $*$♯
	HDLoss	72.90 ± 0.81	91.88 ± 0.37	3.04 ± 0.21	22.43 ± 0.93
	DFRSLoss	75.59 ± 0.41	92.55 ± 0.06	$\mathbf{2.39 \pm 0.05}$	16.74 ± 0.27
	DHDLoss	73.53 ± 0.83	91.82 ± 0.30	2.74 ± 0.13	18.74 ± 0.96
	CELoss	75.52 ± 0.17	91.91 ± 0.19	2.50 ± 0.03	17.83 ± 0.40
	DLoss	$\mathbf{75.70 \pm 0.04}$	92.04 ± 0.11	2.44 ± 0.02	17.63 ± 0.34
SegNet	FRSLoss	74.21 ± 0.43	$\mathbf{92.20 \pm 0.13}$	2.55 ± 0.03 ♯	17.40 ± 0.25 $*$♯
	HDLoss	71.07 ± 0.23	91.90 ± 0.15	3.43 ± 0.07	25.73 ± 0.56
	DFRSLoss	74.41 ± 0.17	91.97 ± 0.18	$\mathbf{2.54 \pm 0.03}$	$\mathbf{17.18 \pm 0.25}$
	DHDLoss	72.20 ± 0.77	91.91 ± 0.09	2.82 ± 0.07	18.88 ± 0.76
	CELoss	74.37 ± 0.20	92.10 ± 0.16	2.56 ± 0.04	17.94 ± 0.32
	DLoss	$\mathbf{74.86 \pm 0.19}$	91.84 ± 0.10	2.60 ± 0.04	18.61 ± 0.34

Table 4.3 shows the performance on the cell dataset using different losses. FRSLoss and DFRSloss performed better than HDLoss and DHDLoss in DC, PA, ASD, and 95th-percentile of HD. It further verifies that the proposed boundary-wise losses are superior to Hausdroff-distance-based losses and achieve higher stability in training segmentation models.

Table 4.3 also depicts the experimental results for the pixel-wise loss and region-wise loss in the UNet, FCN, and SegNet models. It can be seen that the FRSLoss achieves better performance than CELoss and DLoss in the boundary-wise measures (measured by ASD and 95th-percentile of HD), which indicates that the proposed FRSLoss pays more attention to the boundaries than the pixel-wise and region-wise losses. As the same with the Nuclei dataset, the Wilcoxon sign rank test with $p < 0.05$ also is adopted to explore if there is a statistically significant difference between FRSLoss, CELoss and DLoss. Experimental results in Table 4.3 show that FRLoss has a statistically significant difference with CELoss and DLoss in 95th-percentile of HD for all three semantic segmentation models, while for the ASD boundary metric, FRSLoss has a statistically significant difference with DLoss on the UNet and SegNet models, with CELoss on the FCN model. For DC and PA, no statistical difference between the proposed method and DLoss/CELoss. Figure 4.5 uses boxplots to compare the ASD and 95th-percentile of HD of FRSLoss, CELoss and DLoss. From the boxplots, FRSLoss had closer boundary distance than CLoss and DLoss.

Figure 4.6 shows some segmentation results for the three semantic segmentation models and two datasets. Figure 4.6 (a)(b) are the original image and the corresponding label image, respectively. From Figure 4.6 (b), the nuclei dataset has more complicated segmentation boundaries and the boundaries appear as various shapes compared with the cell dataset. Figure 4.6 (c)(e) are the predicted segmentation images of the proposed FRSLoss and DFRSloss. Figure 4.6 (d)(f)(g)(h) are the predicted segmentation images based on HDLoss, DHDLoss, CELoss, and DLoss, respectively. As shown in Figure 4.6, the semantic segmentation models with FRSLoss yield more precise boundaries. Although the evaluation indexes of segmentation performance are nearly the same for FRSLoss, CELoss, and DLoss, the predicted segmentation images in Figure 4.6 show that the boundary of CELoss is relatively blurry and the boundary of DLoss is not as

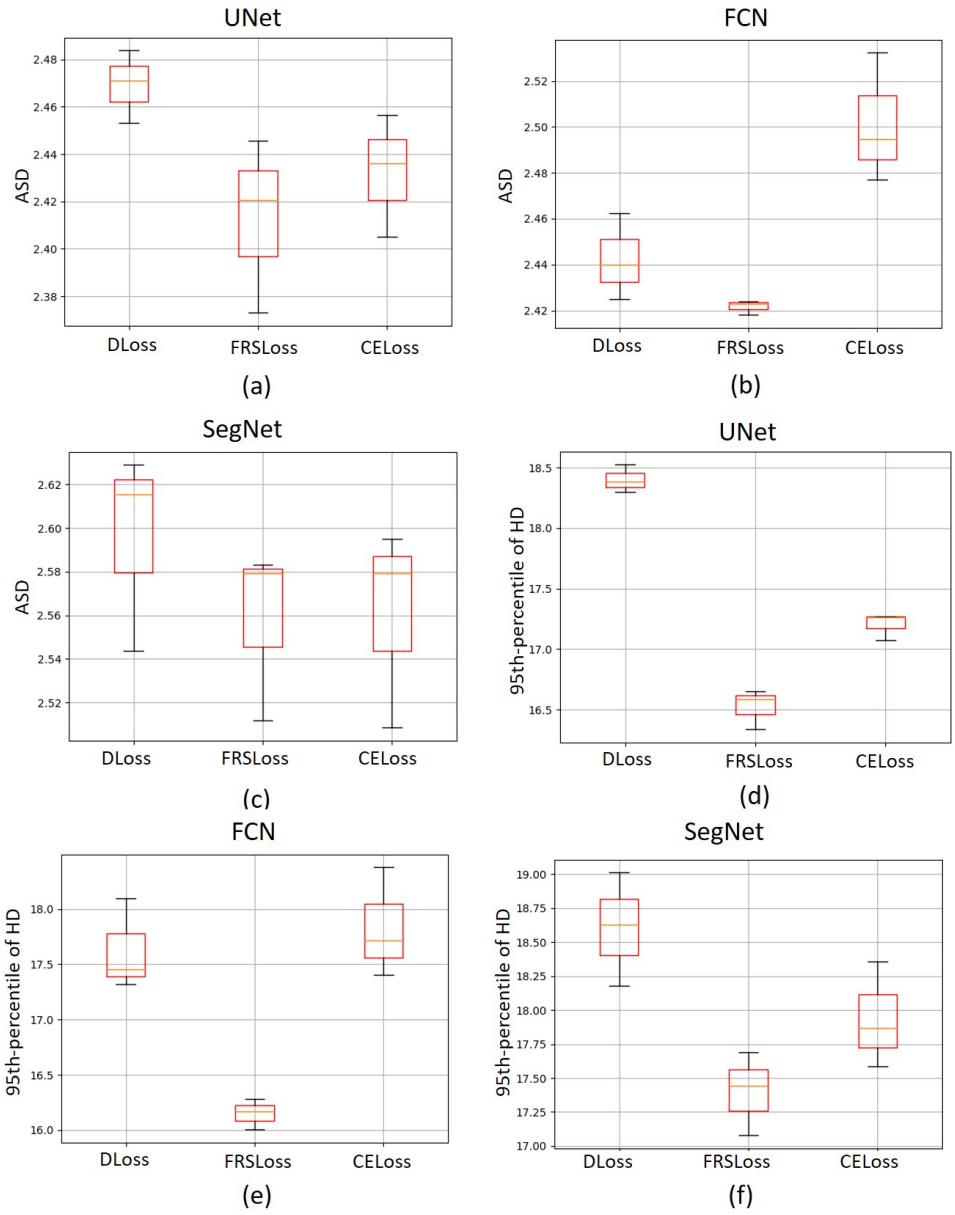


Figure 4.5: Boundary distance metrics ASD and 95th-percentile of HD for DLoss, FRSLoss, and CELoss on Cell dataset. When the values of ASD and 95th-percentile of HD are lower, the predicted image and the ground truth image would have closer boundary distance.

smooth as that of FRSLoss (The red circles and arrows in Figure 4.6 represent the boundary difference between FRSLoss, CELoss and DLoss). It suggests that the boundary-wise loss pays much attention to the boundary, resulting in FRSLosss achieving more accuracy and distinct boundaries than CELoss and DLoss.

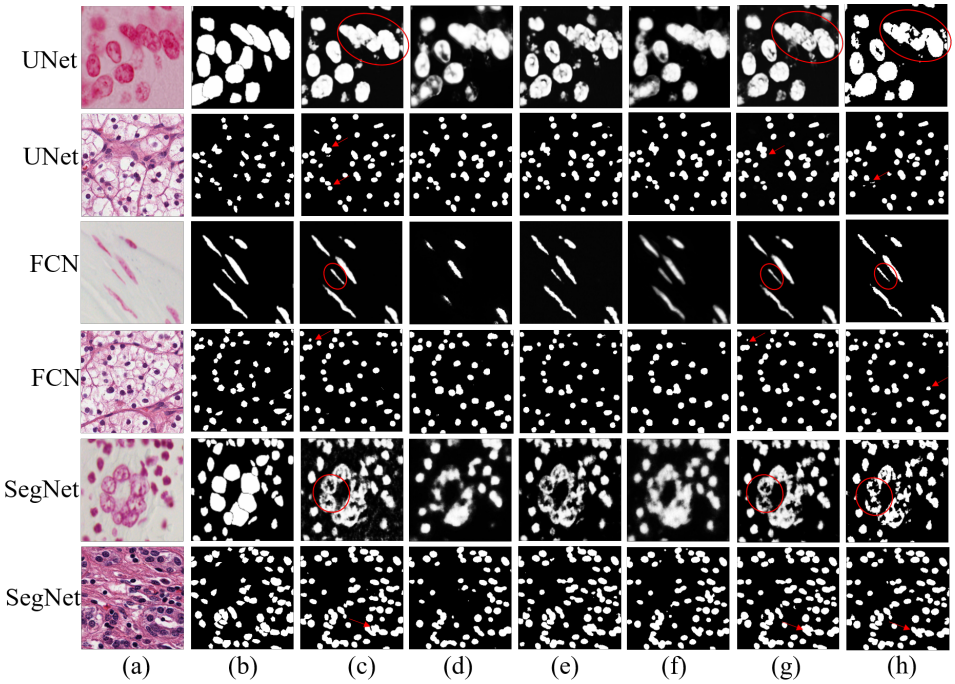
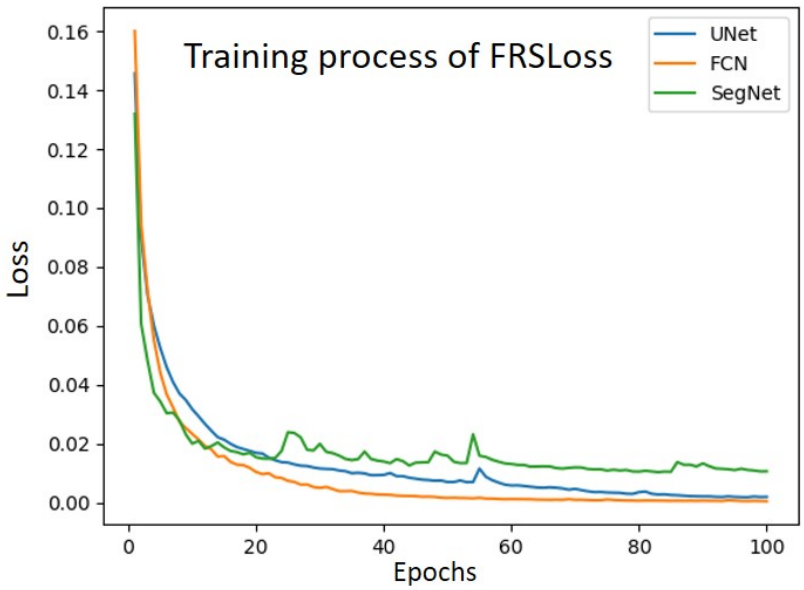


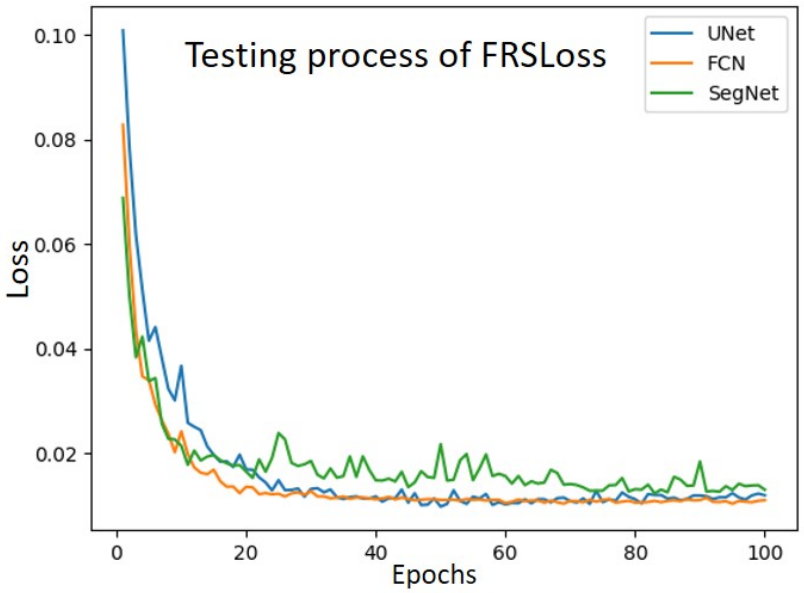
Figure 4.6: Segmentation results for two datasets and three models: (a) original image; (b) benchmark image; (c) predicted segmentation image of FRSLoss; (d) predicted segmentation image of HDLoss; (e) predicted segmentation image of DFRSLoss; (f) predicted segmentation image of DHDLoss;(g) predicted segmentation image of CELoss; (h) predicted segmentation image of DLoss. The red circle and arrow represent the boundary difference between FRSLoss, CELoss, and DLoss.

4.4 Discussion

Table 4.1 and Table 4.3 show that the proposed novel boundary-wise losses, FRSLoss and DFRSLoss, have the ability to enhance the segmentation performance considerably compared with the other boundary-based losses, HDLoss and DHDLoss. To further explore the differences between FRSLoss and

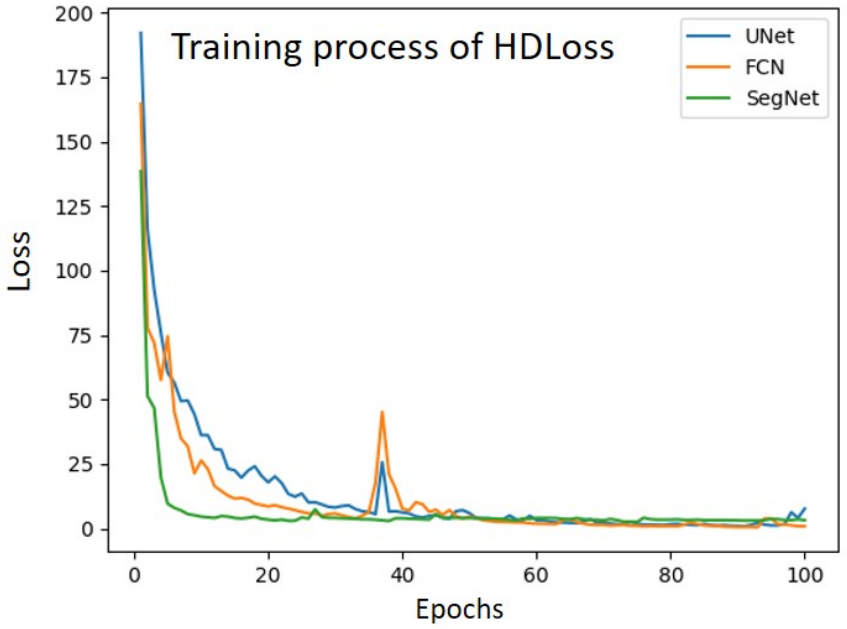


(a) Training loss for FRSLoss

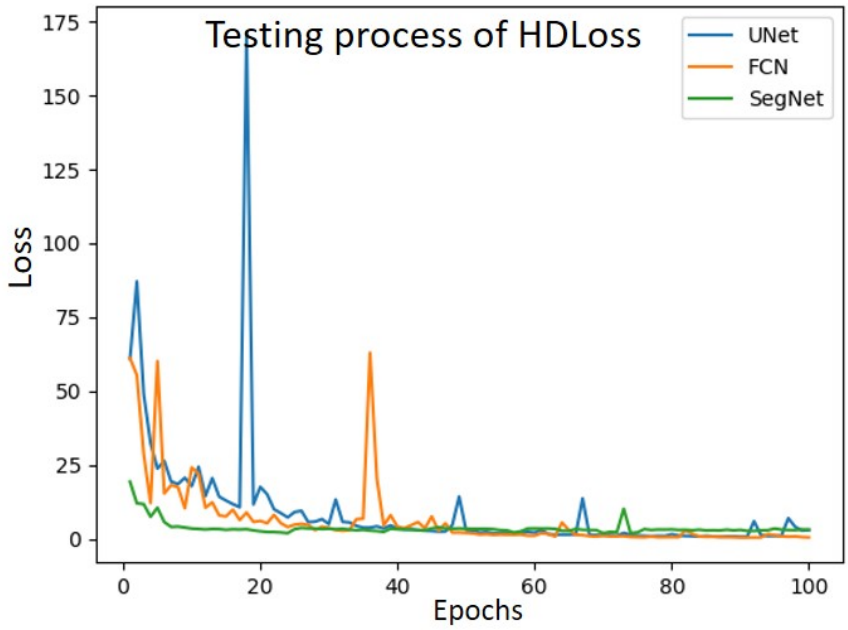


(b) Testing loss for FRSLoss

Figure 4.7: Loss variation curves for UNet, FCN and SegNet

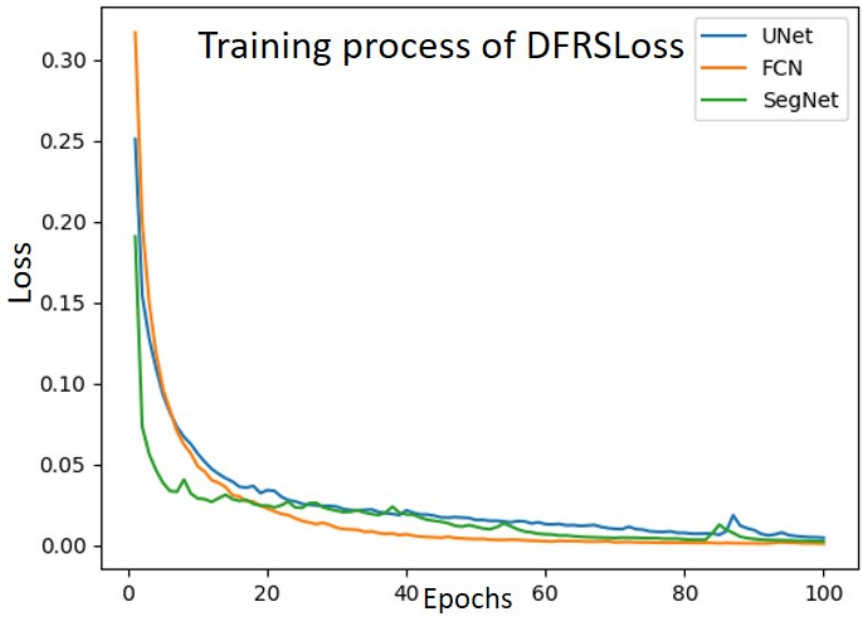


(a) Training loss for HDLoss

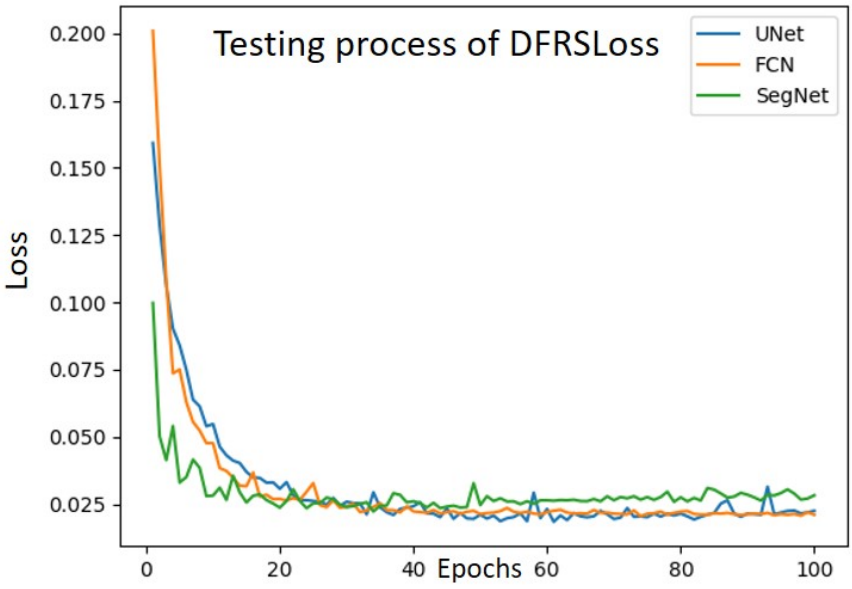


(b) Testing loss for HDLoss

Figure 4.8: Loss variation curves for UNet, FCN and SegNet

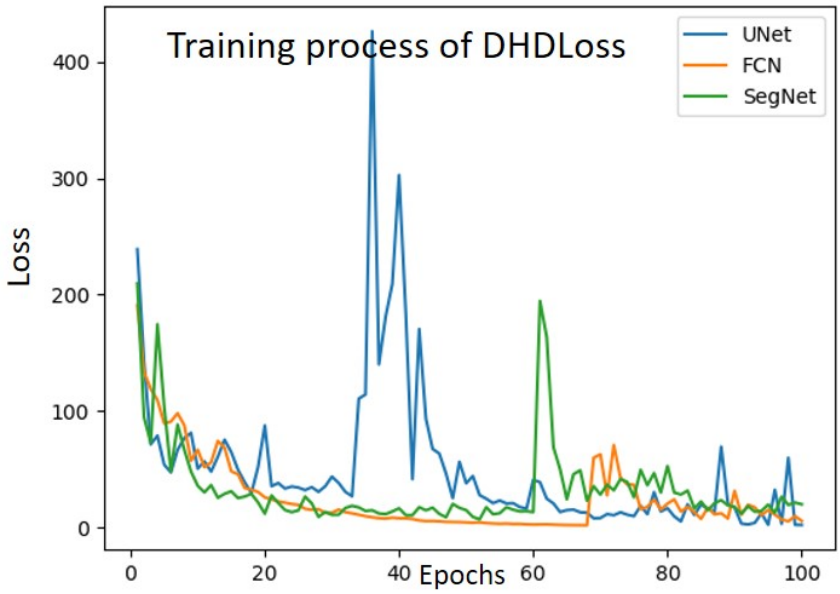


(a) Training loss for DFRSLoss

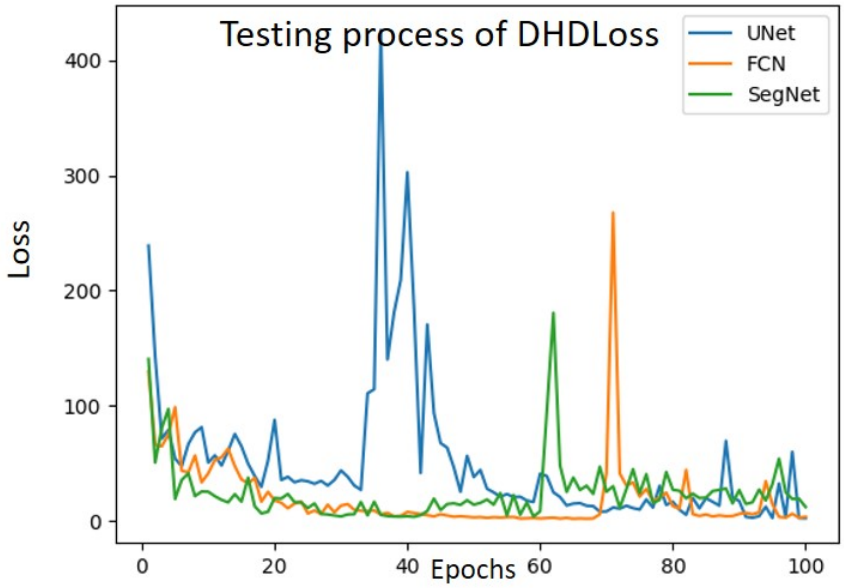


(b) Testing loss for DFRSLoss

Figure 4.9: Loss variation curves for UNet, FCN and SegNet



(a) Training loss for DHDLoss



(b) Testing loss for DHDLoss

Figure 4.10: Loss variation curves for UNet, FCN and SegNet

HDLoss, the training process and testing process for the nuclei dataset are visualized. Figure 4.7 — Figure 4.10 depict the variation curves of the training and testing processes for FRSLoss, HDLoss, DFRSLoss, and DHDLoss. Blue curves mean the UNet model, orange curves refer to the FCN model and green curves refer to the SegNet model. As shown in Figure 4.7 — Figure 4.10, the training process and testing process for HDLoss and DHDLoss fluctuate violently, while the training process and testing process for FRSLoss and DFRSLoss are more stable. Moreover, the range of FRSLoss and DFRLoss values is from 0 to 1, while the range of HDLoss and DHDLoss values is from 0 to infinity. The aforementioned differences in Figure 4.7 — Figure 4.10 are due to the fact that there is a Gaussian function in the formula of FRSLoss and DFRLoss, which plays an important role in normalizing the distance. There are two merits for the normalization operator: 1) it excludes the influence of outliers and makes the gradient descent process more stable and robust (shown in Figure 4.7 — Figure 4.10); 2) it accelerates the convergence rate and reduces the training time (shown in Table 5.2). Thus, the proposed novel boundary-wise losses, FRSLoss and DFRSLoss are more stable and efficient than the other boundary-based losses, HDLoss and DHDLoss.

Compared Table 4.1 with Table 4.3, the proposed FRSLoss and DFRSLoss perform better on the nuclei dataset than that on the cell dataset. The segmentation performance difference between the proposed boundary-wise losses and other boundary-wise losses on the nuclei dataset is more than four times than that on the cell dataset. The reason is that the nuclei dataset has more complicated boundaries than the cell dataset. From Figure 4.6, the boundaries of the nuclei dataset have various shapes while the boundaries of the cell dataset are relatively simple and uniform. The primary merit of the proposed boundary-wise loss is that it pays more attention to the boundary compared with other types of losses. Therefore, FRSLoss and DFRSloss are more beneficial to be applied to objects

with complicated boundaries.

Furthermore, Table 4.1 and Table 4.3 show that the proposed loss concentrates more on the boundaries than the pixel-wise and region-wise losses. In many practical applications, for instance, 3D multi-class image segmentation problems, one single loss is generally unable to obtain a satisfying segmentation result. The popular solution is to integrate the pixel-wise loss, region-wise loss, and boundary-wise loss. In this way, the semantic segmentation models are capable of focusing the pixel, region, and boundary simultaneously, which helps to improve the segmentation performance. Therefore, a stable and robust boundary-wise loss is of considerable importance. According to the experimental results, the proposed boundary-wide losses, FRSLoss and DFRSLoss, are superior to the other boundary-wide losses, HDLoss and DHDLoss, and have the potential to compete with commonly used pixel-wise and region-wise losses, which indicates that the proposed boundary-wise loss is more suitable to combine with the pixel-wise loss, region-wise loss than other boundary-wise losses.

4.5 Summary

In this chapter, a novel boundary-wise loss namely FRSLoss that can be used in various semantic segmentation models is proposed to address objective 2. The FRSLoss derives from the lower approximation of fuzzy rough sets. This is the first time that fuzzy rough sets are incorporated in deep learning models as a loss function for image segmentation. In the proposed FRSLoss formula, a Gaussian kernel is applied to normalize the boundary of the predicted segmentation and the ground truth segmentation, which plays a significant role in stabilizing the training process and saving computational time. Considering the non-convex

nature of the lower approximation of fuzzy rough sets, the distance transform algorithm is utilized to calculate the FRSLoss in semantic segmentation tasks. Moreover, the extension of the proposed FRSloss to multi-class semantic segmentation is also investigated and discussed in this chapter, which broadens the application range further. The experiments with various segmentation models and datasets have verified that the proposed fuzzy rough sets loss is superior to other boundary-wise losses in terms of segmentation accuracy and time complexity. Compared with the commonly used pixel-wise and region-wise losses, the proposed boundary-wise loss has a similar performance but pays more attention to the boundaries.

After completing the training process with the proposed novel FRSLoss function, the next step for semantic segmentation is to use the pre-trained segmentation model to segment new images. One challenge is that there are no ground truth images to quantify the segmentation quality in the real-world application of semantic segmentation models. Therefore, it is crucial to design a quality quantification algorithm to infer the image-level segmentation performance and improve the credibility of semantic segmentation models. In the next chapter, a novel quality quantification algorithm based on fuzzy uncertainty will be proposed to quantify the quality of the predicted segmentation results as part of the model inference process without access to the ground truth images.

Chapter 5

A Novel Quality Quantification Algorithm for Semantic Segmentation Based on Fuzzy Uncertainty

Semantic segmentation models have achieved excellent performance in numerous public datasets. However, the practical application of deep semantic segmentation models is limited, especially in clinical settings, due to the lack of reliable information about the segmentation quality. Therefore, it is essential to create a quality quantification algorithm to assess segmentation performance and then raise the credibility of the semantic segmentation models. In this chapter, by addressing objective 3, a novel quality quantification algorithm based on fuzzy uncertainty is proposed to quantify the quality of the predicted segmentation results as part of the model inference process, as presented in our study [91, 92]

Section 5.1 introduces the background and the motivation of the proposed fuzzy-

uncertainty-based quality quantification algorithm. The detailed procedure is described in Section 5.2. Firstly, test-time augmentation and Monte Carlo dropout are applied simultaneously to capture both the data and model uncertainties of the trained image segmentation model. Then a fuzzy set is generated to describe the captured uncertainty with the assistance of the linear Euclidean distance transform algorithm. Finally, the fuzziness of the generated fuzzy set is adopted to calculate an image-level segmentation uncertainty and therefore to infer the segmentation quality. In Section 5.3, extensive experiments using six medical image segmentation applications on the detection of skin lesion, nuclei, lung, breast, cell and brain are conducted to evaluate the proposed algorithm. Section 5.4 discusses the strengths and challenges of the proposed novel quality quantification algorithm. The contributions and conclusion of this chapter are summarized in Section 5.5.

5.1 Background and Motivation

Although many semantic segmentation models have achieved outstanding performance in different medical applications such as skin lesion [11] and lung tumor [12], their clinical adoptions are limited. This is due to the fact that most of these methods only provide a segmentation result of a given image without an indication of the level of confidence, particularly in the image-level. In clinical practice, the indication of image-level segmentation quality could be used as part of an automatic diagnostic pipeline, in which clinicians could focus on more complicated cases with lower confidence scores suggested by the segmentation model.

To address this issue, quality quantification algorithms are designed to evaluate image-level segmentation quality without the access of ground truth labels.

According to the literature in Section 2.2.3, there are three main approaches: registration-based, learning-based and uncertainty-based. As suggested by Abdar et al. [93], uncertainty in machine learning (ML) based methods normally contains model uncertainty and data uncertainty, and it is highly related to the segmentation quality in a well trained ML model. Furthermore, the computation of uncertainty is normally more efficient than the registration-based method and does not require any learning process in comparison of the learning-based method. Thus it is a natural idea to adopt uncertainty measures to infer the segmentation quality. Herein, the uncertainty-based quality quantification algorithm is primarily discussed, which is more computationally efficient and easier to be implemented in comparison to the other two approaches.

Note that there is a fundamental difference between the predictive probability generated from a segmentation model and the uncertainty measures that we are trying to estimate. Probability provides the likelihood of an event happening (e.g. a pixel being foreground or background in the case of image segmentation). In contrast, uncertainty measures the reliability of the current prediction for a given image. This is normally achieved using Monte Carlo simulation by repeating the prediction process several times and measuring the reliability of the prediction by investigating all Monte Carlo sampling results.

Currently, different uncertainty-based quality quantification methods have been proposed, including IoU [48], Dpw [47], VC [49], Ulabelled [26] and CNNurp [50]. However, these methods have limited application scenarios, which are either sensitive to the data sampling process or suffer from overfitting problem (detailed discussion is provided in Section 5.4). Moreover, all the aforementioned methods only consider the model uncertainty and ignore the data uncertainty, which may cause inaccurate evaluation of segmentation quality. As both the overfitting issue of a trained segmentation model (model uncertainty) and different distributions of clinical data caused by image acquisition variations

(data uncertainty) may potentially lead to a poor-quality segmentation result on new data.

In this chapter, considering fuzzy sets [27] are particularly useful and commonly used to address uncertainty issues and quantify uncertainty, a fuzzy-uncertainty-based quality quantification algorithm is proposed based on fuzzy sets in an attempt to deal with data uncertainty and model uncertainty simultaneously. Moreover, as different regions have different uncertainties and the central region has a lower uncertainty than the boundary region [94], the proposed method utilizes the relationship between the uncertainty and the distance from the pixel's location to the segmentation boundary to generate a fuzzy set. Then the entropy of the fuzzy set is calculated to provide a single overall assessment of the uncertainty in an image, therefore inferring the segmentation quality. Fuzziness is a widely-used measure of uncertainty for a fuzzy set, and besides entropy, there are many other methods available to calculate fuzziness[95] [96]. The proposed novel fuzzy-uncertainty-based quality quantification algorithm addresses objective 3 in this thesis and the detailed description of this algorithm is presented as follows.

5.2 A Novel Quality Quantification Method

The overall pipeline of the proposed fuzzy-uncertainty-based quality quantification algorithm is shown in Figure 5.1. Firstly, a DCNN-based image segmentation model is trained based on a set of training images with their corresponding annotated segmentation masks. Secondly, test-time augmentation and MCdropout are applied in the model inference process to capture the data uncertainty and model uncertainty simultaneously. In this process, N predicted segmentation images $\{Y_1, Y_2, \cdots, Y_N\}$ are produced. Then a distance transform

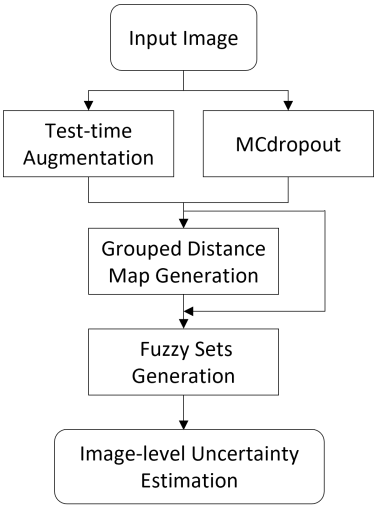


Figure 5.1: The flow chart for the fuzzy-uncertainty-based quality quantification algorithm. It consists of four steps: (1) uncertainty estimation using MCdropout and test-time augmentation; (2) distance map generation; (3) fuzzy-set generation (4) Image level uncertainty estimation.

algorithm is applied on the N predicted images to generate a grouped distance map. This distance map helps in dividing the pixels into groups that are dependent on the distances to the object boundaries. Based on the grouped distance map, a fuzzy set is formalized to describe the N predicted images. Then a fuzziness formula of the fuzzy set is used to calculate the final image-level uncertainty value, therefore to infer the segmentation quality. The detailed descriptions for the key parts of the method are provided in the following subsections.

5.2.1 DCNN-based Image Segmentation Model

In this chapter, UNet [1] and VNet [97] are implemented to segment 2D and 3D medical images respectively since their ability of learning multi-resolution features makes them more suitable for medical images in comparison to other segmentation models.

UNet is constituted by an encoding process, a decoding process and skip connections between them. The encoding process is used to capture the represen-

tative image features of the given input images by using multiple layers of convolutional and down-sampling operations. The decoding process is utilized to up-sample the feature maps so that the predicted segmentation mask having the same image size as the input image. The skip connection helps to transmit the features captured in the encoding process to the decoding process, which is beneficial for the improvement of segmentation performance.

VNet, which is designed to deal with 3D medical images, has the same structure as UNet: a contracting path, an expansive path, and a skip connection. The difference between VNet and UNet is that VNet introduces residuals to tackle the vanishing gradient problem and uses convolutional operators to replace the pooling operators.

The detailed parameter settings for UNet and VNet are described in Section 5.3. Note that the proposed method can be easily applied to other DCNN-based segmentation models.

5.2.2 Test-Time Augmentation and Monte Carlo Dropout

The general process of uncertainty-based quality quantification methods consists of two steps: generate several predicted images for a given test image and then use a reasonable algorithm to quantify the uncertainty existing in the predicted images. Herein, the Test-time augmentation (TTA) and Monte Caro Dropout (MCdropout) are applied simultaneously to capture data uncertainty and model uncertainty by generating N predicted images.

TTA refers to the application of data augmentation during test time and was originally proposed to improve the performance of deep learning models [98]. In 2019, Wang et al. [99] proved the feasibility and rationality of TTA in handling data uncertainty. They used different transformations in TTA to simulate

data uncertainty.

Let $f(\cdot)$ represent the pre-trained DCNN-based image segmentation model, and ω refer to learnable parameters of this model. Given an input image X, the predicted image Y is inferred by:

$$Y = f(X, \omega). \tag{5.1}$$

Then a transformation operator Λ (e.g. scaling, rotation, flipping) with corresponding parameters κ is applied on a test image X to obtain its augmented image $X_a = \Lambda_\kappa(X)$. For example, when the transformation operator Λ is rotation, the κ can be choose from $(0, 2\pi)$.

Next the augmented image X_a is sent into the segmentation model and its related output is calculated by:

$$Y_a = f(X_a, \omega) = f(\Lambda_\kappa(X), \omega). \tag{5.2}$$

In order to get the predicted image of X, the reverse transformation operator Λ_κ^{-1} is applied to Y_a:

$$Y = \Lambda_\kappa^{-1}(Y_a) = \Lambda_\kappa^{-1}(f(\Lambda_\kappa(X), \omega)) \tag{5.3}$$

For example, if the Λ_κ refers to rotate 90 degrees clockwise, Λ_κ^{-1} means to rotate 90 degrees counterclockwise.

Therefore, based on the equation (5.3), for each augmentation operator $\Lambda_{\kappa i}$, the segmentation model produces a predicted image Y_{TTA}^i:

$$Y_{\mathrm{TTA}}^i = \Lambda_{\kappa i}^{-1}(f(\Lambda_{\kappa i}(X), \omega)) \tag{5.4}$$

TTA applies transformation operators to the input image and inverse transformation operators to the output image to capture the data uncertainty. Then, Monte Carlo Dropout (MCdropout) is used to capture the model uncertainty, as demonstrated and proven by Gal and Ghahramani [100].

The implementation of MCdropout is straightforward by randomly dropping some neurons of the segmentation model during the test time. It implies that the parameters of the segmentation model are different in each run. Thus, for the new $\Phi_i(\omega)$, where Φ_i denotes dropping-neurons-related operator in the ith run, the segmentation model produces a predicted image Y_{MC}^i:

$$Y_{\mathrm{MC}}^i = f(X, \Phi_i(\omega)) \tag{5.5}$$

With the combination of equation (5.3) and (5.5), for each pair $(\Lambda_{\kappa i}, \Phi_i(\omega))$, the predicted image is

$$Y_i = \Lambda_{\kappa i}^{-1}(f(\Lambda_{\kappa i}(X), \Phi_i(\omega))). \tag{5.6}$$

The equation (5.6) presents the computational process of the predicted image by using TTA and MCdropout simultaneously. Therefore, given a test image X, when choosing N different pairs $\{(\Lambda_{\kappa 1}, \Phi_1(\omega)), (\Lambda_{\kappa 2}, \Phi_2(\omega)), \cdots, (\Lambda_{\kappa N}, \Phi_N(\omega))\}$, the segmentation model generates N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$. These N predicted images are then used for subsequent uncertainty estimation.

5.2.3 Grouped Distance Map Generation

After using TTA and MCdropout simultaneously to capture the data uncertainty and model uncertainty, N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$ are generated. The next step is to calculate a grouped distance map, which aims to divide the seg-

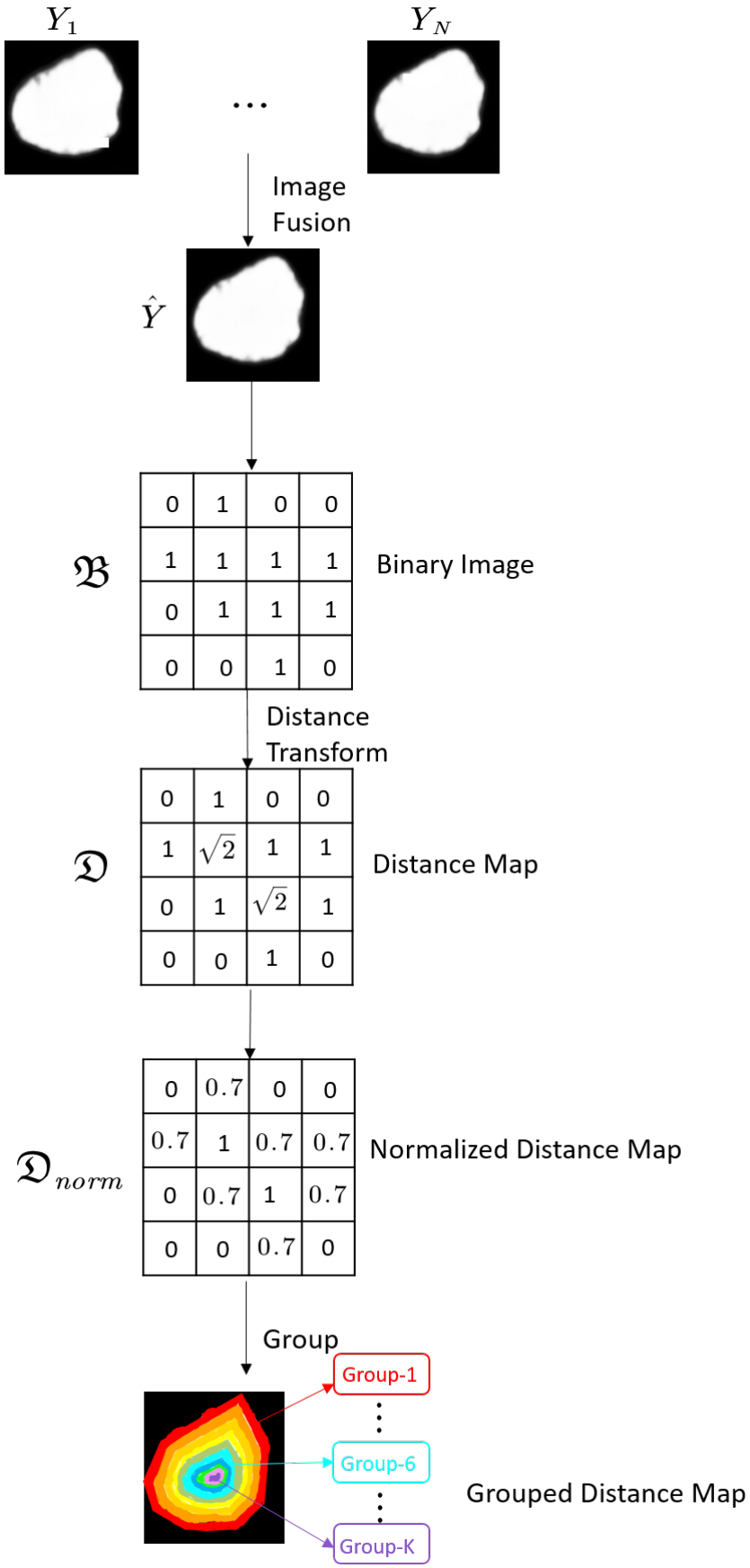


Figure 5.2: The pipeline to calculate the grouped distance map from predicted images obtained with test-time augmentation and MCdropout.

mented target area into K groups based on the pixel distance to the segmentation object boundary. This is based on the assumption that the further a pixel is from the boundary, the lower the uncertainty is in its segmentation, as previously suggested by Nair et al. [94].

Figure 5.2 shows the detailed steps to calculate the grouped distance map. Firstly, an image fusion operator is applied on the N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$ to obtain the fused image Y. Herein, two different image fusion methods are implemented and compared. One is the widely-used average fusion method which is expressed as $\hat{Y} = \frac{1}{N}\sum_{i=1}^{N} Y_i$.

The other one is fuzzy fusion method, which is inspired by Diamantis and Iakovidis [101] and was initially used in the pooling layers of DCNN models. This is the first time that the fuzzy fusion method is used to fuse N predicted images in the inference process. The fuzzy fusion method uses three fuzzy membership functions μ_1, μ_2 and μ_3 as shown in Figure 5.3 to map the predicted pixel values to membership values according to their value ranges (low, medium, and high). The corresponding formulas for μ_1, μ_2 and μ_3 are represented as:

$$\mu_1\left(p_{k,j}^i\right) = \begin{cases} 1 & p_{k,j}^i \leq c \\ \frac{d-p_{k,j}^i}{d-c} & c < p_{k,j}^i \leq d \\ 0 & p_{k,j}^i > d \end{cases}$$

$$\mu_2\left(p_{k,j}^i\right) = \begin{cases} 0 & p_{k,j}^i \leq e \\ \frac{p_{k,j}^i-e}{f-e} & e < p_{k,j}^i \leq f \\ \frac{g-p_{k,j}^i}{g-f} & f < p_{k,j}^i \leq g \\ 0 & p_{k,j}^i > g \end{cases}$$

$$\mu_3\left(p_{k,j}^i\right) = \begin{cases} 0 & p_{k,j}^i \leq h \\ \frac{p_{k,j}^i - h}{q-h} & h < p_{k,j}^i \leq q \\ 1 & p_{k,j}^i > q \end{cases} \tag{5.7}$$

where $p_{k,j}^i$ is the pixel value in the location (k, j) of the ith predicted image Y_i. This fuzzy membership mapping process helps in determining the majority of the predicted N values for a pixel is low, medium or high. In fact, these three membership functions μ_1, μ_2 and μ_3 essentially act as three filters which are all applied to each pixel. The x-axis is the pixel value and the y-axis is the corresponding membership value. Using the corresponding membership value, μ_1 only allows low pixel values to pass, μ_2 only allows medium pixel values to pass, and μ_3 only allows high pixel values to pass. Based on the experimental results suggested by Diamantis and Iakovidis [101], the seven parameters c, d, e, f, g, h, q for μ_1, μ_2 and μ_3 in Figure 5.3 are determined by the maximum pixel value $p_{\max}$: $c = \frac{p_{\max}}{6}$, $d = \frac{p_{\max}}{2}$, $e = \frac{p_{\max}}{4}$, $f = \frac{p_{\max}}{2}$, $g = \frac{3p_{\max}}{4}$, $h = \frac{p_{\max}}{2}$, $q = \frac{3p_{\max}}{4}$. Note that for predicted images of the semantic segmentation model, the pixel value refers to the probability of belonging to the target segmentation object and its maximum value is 1. Thus in the proposed method, $c = \frac{1}{6}$, $d = \frac{1}{2}$, $e = \frac{1}{4}$, $f = \frac{1}{2}$, $g = \frac{3}{4}$, $h = \frac{1}{2}$, $q = \frac{3}{4}$.

After applying μ_1, μ_2 and μ_3 to the pixels in the N predicted images, the membership values produced by the group with the highest sum of membership values are retained. Next, the center of gravity [102] is applied to calculate the final fusion value based on this selected membership function. Intuitively, this fuzzy fusion process acts as a filtering process to remove the outliers in the N predicted segmentation results, which results in a more reliable predicted image Y. The implementation detail of fuzzy fusion is given in Algorithm 5.1.

Subsequently, a binary function $\mathfrak{B}(p) = \begin{cases} 1, p \geq 0.5 \\ 0, p < 0.5 \end{cases}$ is applied on Y to get

the binary image $\mathfrak{B}$, where p means the pixel value.

Next, the distance map is obtained as follows. Euclidean distance transform algorithm [89] is used to obtain the closest distance from pixels in the segmented object to the object boundary. In detail, the distance map $\mathfrak{D} = f_{DT}(\mathfrak{B})$, where $\mathfrak{D}$ and $\mathfrak{B}$ have the same size and the algorithm f_{DT} used to calculate the pixel value in $\mathfrak{D}$ is presented in Section 4.2.3. Note that this distance transform algorithm can be applied to various different shapes (e.g. convex, concave, and hollow) and multiple objects.

Then, to ensure the distance map is comparable across different images and objects, the Min-Max scaling algorithm is applied to normalize the distance map using $\mathfrak{D}_{\text{norm}} = \frac{\mathfrak{D}-\mathfrak{D}_{\text{min}}}{\mathfrak{D}_{\text{max}}-\mathfrak{D}_{\text{min}}}$. In this way, the pixel distances are normalized to the range of $[0,1]$. Finally, the normalized pixel distances in $\mathfrak{D}_{\text{norm}}$ are divided into K groups $\{g_1, g_2, \cdots, g_K\}$ with evenly distributed distances from 0 to 1. Specifically, if the pixel distance is in the range of $[\frac{(t-1)}{K}, \frac{t}{K})$, where $t \in \{1,2,\cdots,K\}$, this pixel is assigned to the group g_t. Pixels with the normalized distance of 1 are assigned to the Kth group.

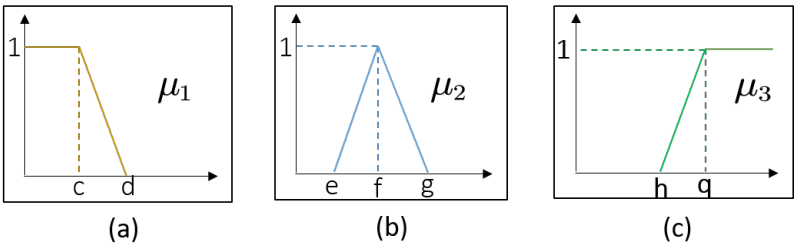


Figure 5.3: Three membership functions, (a) maps the small pixel values to high membership values, (b) maps the medium pixel values to high membership values, (c) maps the large pixel values to high membership values.

5.2.4 Fuzzy Sets Generation

Having the grouped distance map generated, the next step is to generate a fuzzy set to describe the N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$. In this way, the segmen-

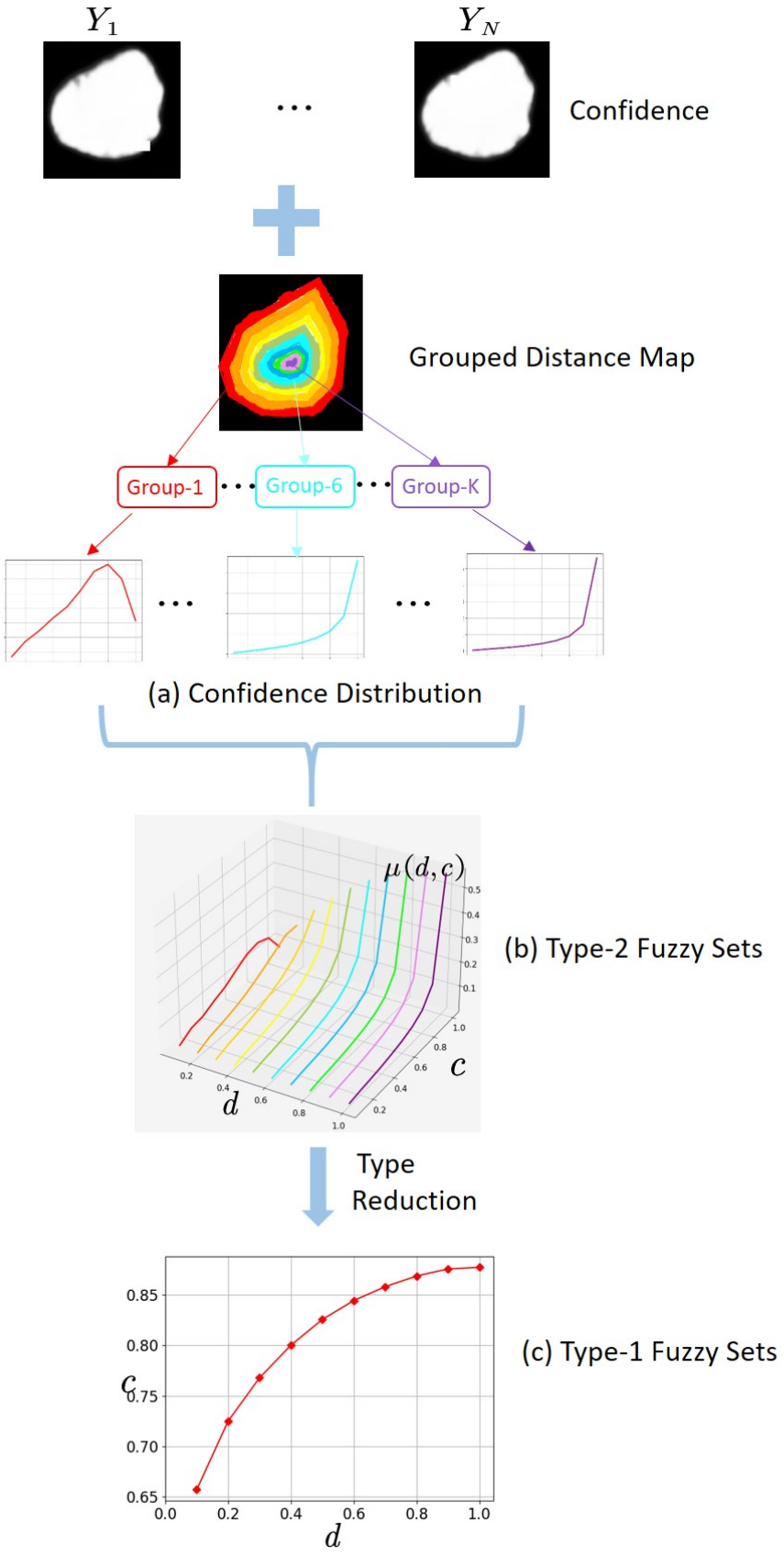


Figure 5.4: The pipeline to obtain the fuzzy sets

Algorithm 5.1 Fuzzy Image Fusion

Input: N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$, the size of each predicted image is $M \times M$

Output: Fused Image $\hat{Y}$

given $I = [Y_1, Y_2, \cdots, Y_N]$, the size of I is $N \times M \times M$
choose three membership functions μ_1, μ_2, μ_3, the corresponding function curves are given in Figure 5.3
for i from 1 to M **do**
    for j from 1 to M **do**
        calculate the membership value
            $V_1 = \sum_{k=1}^{N} \mu_1 (I(k,i,j))$
            $V_2 = \sum_{k=1}^{N} \mu_2 (I(k,i,j))$
            $V_3 = \sum_{k=1}^{N} \mu_3 (I(k,i,j))$
        $t = \text{argmax}\{V_t\}$, where $t \in \{1,2,3\}$
        obtain the fusion value for pixel at location i, j
        $\hat{Y}(i,j) = \frac{\sum_{k=1}^{N} \mu_t (I(k,i,j)) \times I(k,i,j)}{V_t}$
return $\hat{Y}$

tation uncertainty is able to be calculated by estimating the fuzzy uncertainty of the fuzzy set. The process is illustrated by Figure 5.4.

It is observed that the predicted pixel-wise confidence values of a segmentation model are positively correlated to their distance to the predicted object boundary. In other words, the further away a pixel is from the object boundary the more confident the prediction is, as illustrated in Figure 5.4(c). This curve can be considered as a type-1 fuzzy set.

On the other hand, as the predicted images and the grouped distance map have been obtained, it is possible to get such a type-1 fuzzy set described above by following the steps below.

From the previous process, it is known that the pixel distances in grouped distance map are divided into K groups $\{g_1, g_2, \cdots, g_K\}$. As each value in the N predicted images represents the confidence level of the pixel belonging to the target segmentation object, these values can be called confidence values. Thus, for each group g_t, according to distance from the pixel's location to the segmen-

tation object boundary, we obtain the confidence values $\{c_1^{t,1}, \cdots c_\sigma^{t,1}, c_1^{t,2}, \cdots, c_\sigma^{t,2}, \cdots, c_1^{t,N}, \cdots c_\sigma^{t,N}\}$ from the N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$, where σ refers to the number of pixels belonging to the group g_t for each predicted image. Then the distribution of the confidence values for each group g_t is obtained as presented in Figure 5.4(a). It is expected that the central object region (i.e. larger distance to the boundary) contains more confident pixels and vice versa. The confidence distribution can be regarded as the secondary membership function due to the fact that the confidence value is a continuous function rather than a single value. By the combination of all the distributions, a 3D distribution plot is generated as shown in Figure 5.4(b), which can be treated as a type-2 fuzzy set. The primary variable (x-axis) is the distance from the pixel to the segmentation boundary, and the secondary variable (y-axis) is the confidence value to represent whether this pixel belongs to the target segmentation class.

In the next step, to get the type-1 fuzzy set, a type reduction method is applied to the type-2 fuzzy set. An efficient method, known as the centroid method (i.e. weighted average in formula (5.8)), is used to convert the distribution (the secondary membership function) of each group to a single confidence level.

$$\frac{\sum_i c_i \mu_i(d, c_i)}{\sum_i \mu_i(d, c_i)} \tag{5.8}$$

Note that this is not a standard type reduction method for type-2 fuzzy sets. However, it can be considered as an extension (or a variation) of the Nie-Tan type reduction operator [103]. Hence, in this chapter, this centroid method is named as a type reduction method. After applying the type reduction, a type-1 fuzzy set (as illustrated in Figure 5.4(c)) is successfully obtained.

5.2.5 Image-level Uncertainty Estimation

After obtaining the type-1 fuzzy sets, fuzzy uncertainty is calculated based on the fuzziness of fuzzy sets. Inspired by Pal et al. [95] and Wu et al. [96], the fuzziness of fuzzy sets is the quantification of uncertainty for a given type-1 fuzzy set. Fuzzy entropy E is commonly used to measure the fuzziness and has the following attributes: 1) $E(A) = 0 \Longleftrightarrow \mu_A(x) = 0$ or $1 \; \forall \, x \in \mathbb{U}$; 2) $E(A)$ is maximum $\Longleftrightarrow \; \mu_A(x) = 0.5 \; \forall \, x \in \mathbb{U}$; 3) $E(A) = E(1-A)$, where $\mu_{1-A}(x) = 1 - \mu(A) \; \forall x \in \mathbb{U}$; where A refers to fuzzy sets and μ means the corresponding membership function. In this chapter, the following fuzzy entropy formula is chosen to calculate the uncertainty.

$$U(A) = 1 - \frac{\left[\sum_{i=1}^{K} |2\mu_A(x_i) - 1|^2\right]^{\frac{1}{2}}}{K^{\frac{1}{2}}}, \tag{5.9}$$

where K is the number of discrete points in the $\mathbb{U}$ and x_i is the i_{th} discrete value, μ_A is membership function for the fuzzy set A.

Through this fuzziness estimation process, the image-level uncertainty is finally generated. A high fuzzy entropy value means the uncertainty of the segmentation result is high.

Finally, Algorithm 5.2 summarizes the overall calculation process of the proposed fuzzy-uncertainty-based quality quantification method.

5.3 Evaluation and Results

In this section, the performance of the proposed algorithm is discussed. Firstly, the setting of the two hyper-parameters in the proposed method was experimentally determined. Then, the experiments of calculating the correlation be-

Algorithm 5.2 Fuzzy-uncertainty-based Quality Quantification

Input: raw image X and its size is $M \times M$
Output: image-level uncertainty value U

apply TTA and MCdropout to obtain N predicted images $\{Y_1, Y_2, \cdots, Y_N\}$
obtain the grouped distance map $\mathfrak{D}_{\text{norm}}$ (Section 5.2.3)
divide the $\mathfrak{D}_{\text{norm}}$ into K groups $g_1, g_2, ..., g_K$
for i from 1 to K **do**
    define the confidence group C_i, where the size of C_i is $s = 0$
    for j from 1 to N **do**
        for each pixel p_t^j in Y_j **do**
            if $g_i \leq \mathfrak{D}_{\text{norm}}(p_t^j) < g_{i+1}$ **then**
                add p_t^j into C_i, and the size of C_i is updated by $s \leftarrow s + 1$
    calculate the distribution of C_i
    obtain the type-1 fuzzy set membership value using type reduction operator (Section 5.2.4)
calculate the uncertainty value U based on equation (5.9)
return U

tween segmentation quality (measured by the Dice coefficient) and the image-level uncertainty (measured by different quality quantification methods) were conducted. The proposed method was compared with several state-of-the-art methods. Finally, the application of uncertainty for quality quantification was explored. Datasets, implementation details, experimental methods, and experimental results are given in the following subsections.

Six public medical image datasets are used in the experiments to verify the performance of the fuzzy-uncertainty-based quality quantification method.

- Skin lesion: this dataset [11] includes 2594 raw dermoscopic 2D images with the corresponding ground truth images annotated by dermatologists. The target of this dataset is to segment skin lesions from complicated dermoscopic images.

- Nuclei: this dataset comes from the 2018 Data Science Bowl https://www.kaggle.com/c/data-science-bowl-2018-/data and has considerably compli-

cated object boundaries. Each image in this dataset has multiple nuclei to be segmented and these nuclei differ from each other in cell types, magnification and imaging modality. There are 670 raw 2D images with the corresponding ground truth masks.

- Lung [104]: this dataset consists of 704 chest X-ray images and the aim is to extract the lung boundary from the given chest image. Each X-ray image has various sizes and requires resizing to a uniform size before feeding into the semantic segmentation model.

- Breast tumor[105]: the dataset includes 780 breast ultrasound images with the corresponding binary masks. The aim of this dataset is to delineate the breast cancer tumour from breast ultrasound images. The average size of raw images is 500×500 pixels.

- Cell [106]: this dataset consists of 1200 raw simulated HCS cell images with corresponding ground truth images. The size of each image is 696 x 520 pixels. The target of this dataset is to segment the cell areas in order to count the number of cells.

- Brain [107]: this dataset consists of 414 3D magnetic resonance images (MRI). Each image in this dataset has five categories: cortex, subcortical-Gray-Matter, White-Matter, cerebrospinal fluid and background to be segmented.

These datasets are chosen to evaluate the proposed method as they have different properties. Skin lesion and breast tumor datasets only have one object to segment, while nuclei, lung, cell datasets have multiple objects to segment. Furthermore, the segmentation objects in Skin lesion, nuclei, breast tumor, and cell datasets are very flexible that present all sorts of shapes, while the segmentation shapes in the lung dataset are more rigid. Compared to other datasets, the

Brain dataset has numerous categories to segment and each image is in three dimensions.

In the experiments, UNet (2D) and VNet (3D) were used as the baseline segmentation models for all experiments. The pixel size for the input and output of UNet is 256×256, and for that of V-Net is $96 \times 96 \times 96$, which means that all the datasets should be reshaped into the same size. These two models have five layers and are constituted by the encoding process and decoding process. During the encoding stage, the number of kernels for convolutional operators was 16, 16 (first layer); 32, 32 (second layer); 64, 64 (third layer); 128, 128 (fourth layer); and 256, 256 (fifth layer). The kernel size of each layer for UNet was 3×3 with the stride = 1 and maxpooling was adopted to change the resolution of each layer feature map. The kernel size for each layer for VNet was $2 \times 2 \times 2$ with the stride = 2 and the connection structure is similar to a residual network [108]. During the decoding stage, de-convolution operations with the kernel size 2×2 (UNet) and $2 \times 2 \times 2$ (VNet) were adopted to recover the feature map sizes. In the Vnet, convolutional operators were used to replacing the max-pooling operators. Each layer of the UNet and VNet also includes a dropout layer, which is used to prevent overfitting during the training time and capture model uncertainty during the test time. Adam optimization algorithm [109] is used to update the parameters with the learning rate of 10^{-4}.

Pearson correlation coefficient (PCC) [110] was used to measure the performance of different quality quantification methods. It calculates the linear correlation between the uncertainty value (calculated by a quality quantification algorithm) and the image segmentation quality (measured by the Dice coefficient). Note that Dice $= \frac{2|A \cap B|}{|A|+|B|}$, where A refers to the foreground areas of the predicted image and B means the foreground areas of the ground truth image. When the PCC gets closer to -1 or 1, it means that the uncertainty value has a strong linear correlation with the Dice coefficient and is capable of inferring the

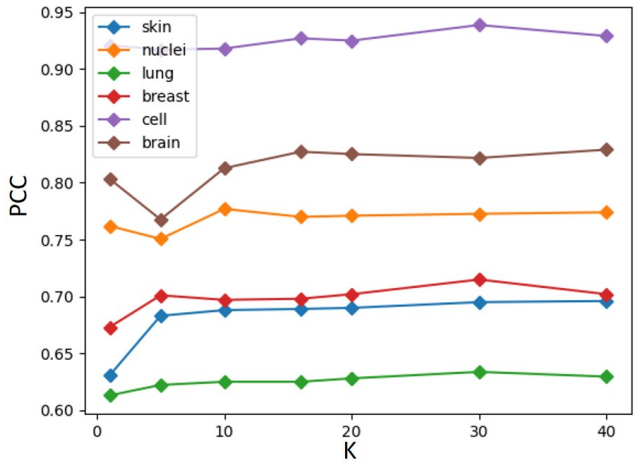
segmentation quality well. A PCC value closer to 0 indicates a poor correlation.

In all experiments, the raw images in each dataset were resized to 256×256 (2D) or $96 \times 96 \times 96$ (3D). Then each dataset was divided into five groups for five-fold cross validation. In each fold, four groups were used for training (80%) and validation (20%), and the remaining group was used for testing. All the experiments ran on a workstation with NVIDIA GeForce GTX1080Ti and i7-3820 CPU. The deep learning framework is PyTorch [111].

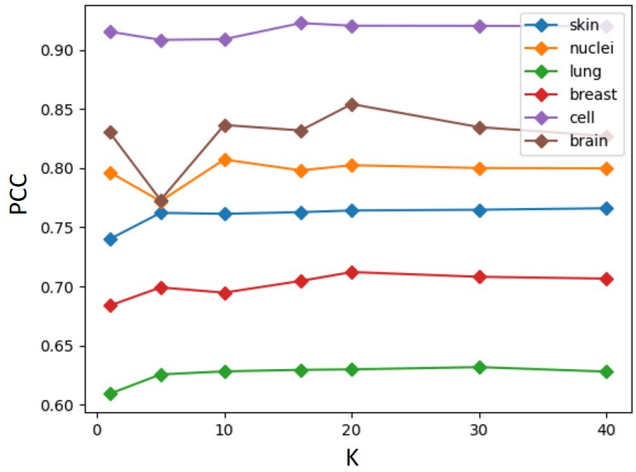
5.3.1 Parameter Settings

The proposed fuzzy-uncertainty-based quality quantification algorithm contains two hyper-parameters: K (the number of the groups in the grouped distance map) and N (the number of the predicted images generated based on TTA and MCdropout). As explained in Section 5.2.2, N is determined by the number of pairs $(\Lambda_\kappa, \Phi(\omega))$. Λ_κ is a combination of flipping, rotation, and scaling operations. In terms of flipping, an image can be horizontally or vertically flipped, or unchanged. For rotation, an image is rotated by r degrees which is randomly chosen from 0 to 2π. For scaling, an image is scaled by a scaling factor s which is a random number between 0.8 and 1.5. On the other hand, for each run, the parameters of the segmentation model ω are different after using the dropping-neurons-related operator Φ since the input units in the dropout layers are randomly set to be 0 at a frequency rate of 0.2 during the test time. In our experiments, TTA and MCdropout were used simultaneously to generate N predicted images by choosing different values of $(\Lambda_\kappa, \Phi(\omega))$. Herein, different hyper-parameter settings have been explored, specifically, the influence of the number of the pairs $(\Lambda_\kappa, \Phi(\omega))$ on the performance.

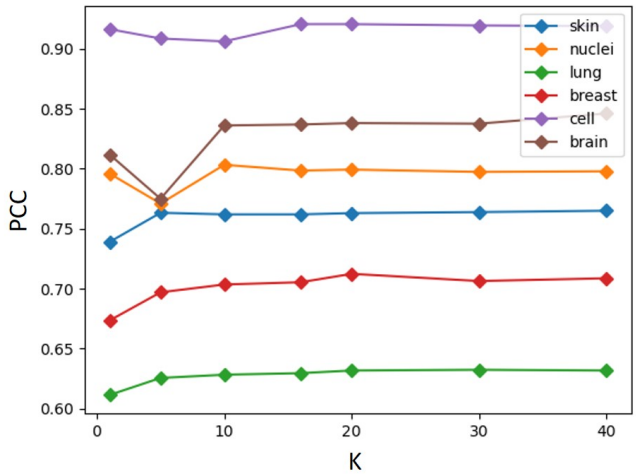
N was chosen from $\{4, 8, 12, 24, \cdots, 96\}$, while K was chosen from $\{1, 5, 10, 16,$



(a) $N = 12$

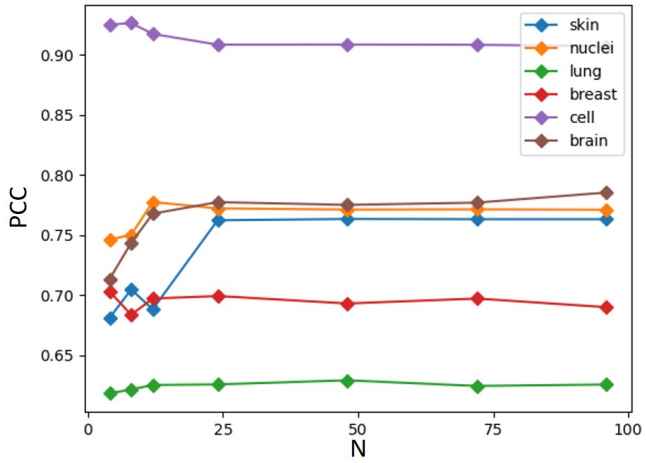


(b) $N = 24$

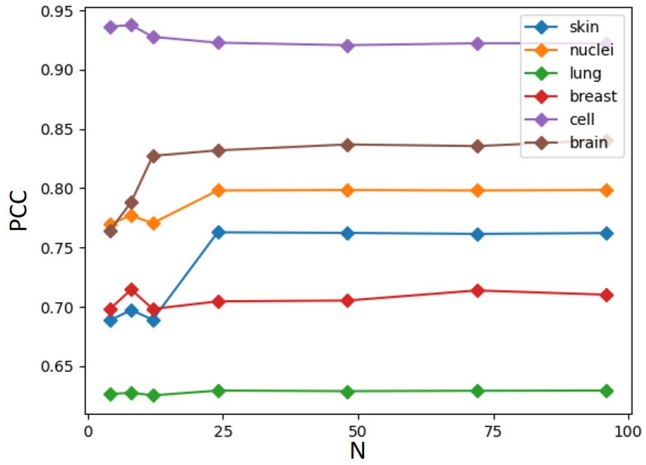


(c) $N = 48$

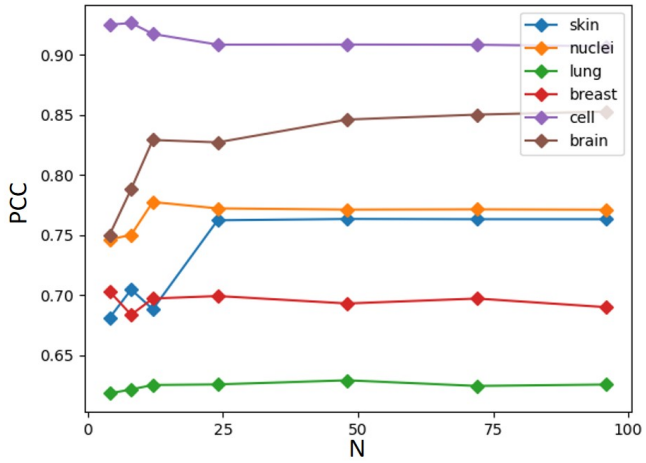
Figure 5.5: The curve of PCC with different K values when N is equal to (a) 12, (b) 24, and (c) 48 respectively



(a) $K = 5$



(b) $K = 16$



(c) $K = 40$

Figure 5.6: The curve of PCC with different N values when K is equal to (a) 5, (b) 16, and (c) 40 respectively

$\cdots, 40\}$. Figure 5.5 shows the curve of PCC between the uncertainty value (calculated by the proposed method) and the image segmentation quality (measured by the Dice coefficient) with different K values when N is equal to 12, 24 and 48 respectively. Figure 5.6 shows the curve for different N values when K is equal to 5, 16 and 40 respectively. Note that in order to make the curves readable, absolute values of PCC are used.

In Figure 5.5, when $K \geq 20$, the performance of all datasets remain stable. This is due to limitation of the resolution of the distance map. A higher resolution in splitting the distance map by using $K \geq 20$ will not make a significant difference. In Figure 5.6, when $N \geq 24$, all datasets achieved a stable performance. Therefore, based on Figure 5.5 and Figure 5.6, in the following experiments, the number of predicted images is set as 24 and the number of groups in the grouped distance map is set as 20.

5.3.2 Comparison of Uncertainty-based Quality Quantification Methods

In this section, the proposed method is compared with other state-of-the-art uncertainty based methods. Fdata, FQC, and FQCfuzzy are our fuzzy-uncertainty-based quality quantification methods, while VC, Ulabelled, IoU, Dpw and CNNurp are other five state-of-the-art uncertainty-based quality quantification methods as mentioned in Section 2.2.3. Fdata only considers the data uncertainty. FQC is the proposed method in this chapter by using the average image fusion method to fuse the predicted images. FQCfuzzy is similar as FQC but using the fuzzy image fusion method (see in Algorithm 5.1).

Table 5.1 shows the Pearson correlation coefficient (PCC) [110] between the uncertainty value (calculated by different quality quantification algorithms) and the

Table 5.1: Experimental results for different uncertainty-based quality quantification methods based on five-fold cross-validation. The mean and standard deviation values of the five results are calculated to represent the Pearson correlation between measured uncertainty and the Dice coefficient in terms of Mean $\pm$ standard deviation.

Methods	Skin	Nuclei	Lung	Breast	Cell	Brain	Avg. Rank
Fdata	$-0.736 \pm 0.027(4)$	$-0.701 \pm 0.071(8)$	$-0.559 \pm 0.029(5)$	$-0.652 \pm 0.191(4)$	$-0.889 \pm 0.023(4)$	$-0.818 \pm 0.001(3)$	4.7
FQCfuzzy	$-0.791 \pm 0.045(1)$	$-0.871 \pm 0.036(1)$	$-0.604 \pm 0.050(3)$	$-0.749 \pm 0.080(1)$	$-0.917 \pm 0.010(2)$	$-0.817 \pm 0.011(4)$	2
FQC	$-0.757 \pm 0.042(3)$	$-0.802 \pm 0.080(3)$	$-0.628 \pm 0.041(1)$	$-0.707 \pm 0.087(2)$	$-0.909 \pm 0.025(3)$	$-0.836 \pm 0.003(1)$	2.2
VC	$-0.782 \pm 0.044(2)$	$-0.727 \pm 0.092(6)$	$-0.558 \pm 0.036(6)$	$-0.667 \pm 0.087(3)$	$-0.880 \pm 0.037(5)$	$-0.826 \pm 0.003(2)$	4.0
Ulabelled	$-0.694 \pm 0.051(5)$	$-0.745 \pm 0.084(5)$	$-0.621 \pm 0.055(2)$	$-0.602 \pm 0.078(5)$	$-0.953 \pm 0.013(1)$	$-0.744 \pm 0.003(7)$	4.2
IoU	$0.536 \pm 0.033(8)$	$0.797 \pm 0.078(4)$	$0.571 \pm 0.029(4)$	$0.496 \pm 0.079(8)$	$0.841 \pm 0.107(6)$	$0.8077 \pm 0.012(6)$	6
Dpw	$0.584 \pm 0.048(7)$	$0.860 \pm 0.083(2)$	$0.557 \pm 0.030(7)$	$0.520 \pm 0.075(7)$	$0.831 \pm 0.176(7)$	$0.814 \pm 0.009(5)$	5.8
CNNurp	$0.632 \pm 0.092(6)$	$0.711 \pm 0.156(7)$	$0.498 \pm 0.155(8)$	$0.598 \pm 0.196(6)$	$0.750 \pm 0.074(8)$	$0.283 \pm 0.055(8)$	7.2

image segmentation quality (measured by the Dice coefficient). The numbers in parentheses indicate the performance ranking of different quality quantification methods for a given dataset. The average rank (Avg. Rank) is calculated by the average performance of a quality quantification algorithm on all six datasets.

From Table 5.1, conclusions could be drawn from three aspects. Firstly, FQC (MCdropout+TTA) outperformed the Fdata (TTA only) method for all six datasets by a large margin, indicating a significant contribution of estimating the model uncertainty using MCdropout.

Secondly, the fuzzy image fusion method (FQCfuzzy) outperformed the FQC method with average fusion in the proposed fuzzy certainty-based quality quantification algorithm. Table 5.1 shows that FQCfuzzy has a higher PCC than FQC on Skin, Nuclei, Breast and cells datasets. Moreover, for the average rank on the entire datasets, FQCfuzzy is 2.0 which is better than 2.2 for FQC, which indicates that the proposed image fusion method could further improve the performance.

Thirdly, the proposed quality quantification algorithm (FQC with the average fusion) performs better than the other five state-of-the-art quality quantification methods: VC, Ulabelled, IoU, Dpw and CNNurp. Note that VC, Ulabelled, IoU, Dpw and CNNurp also use the average image fusion method. Although FQC does not achieve the best performance on all datasets (1st on the Lung, Breast and Brain datasets and 2nd on Skin, Nuclei and Cell datasets), its overall ranking on all selected datasets is the highest, which suggests that FQC is more stable and robust than the other uncertainty-based quality quantification methods.

Figure 5.7 shows the scatter plots for the FQC, VC, Ulabelled, IoU, Dpw and CNNurp methods. Each point represents a test image and different colors refer to different datasets. As shown in the scatter plots, FQC, VC, Ulabelled have a negative correlation with the Dice coefficient, while IoU, Dpw and CNNurp

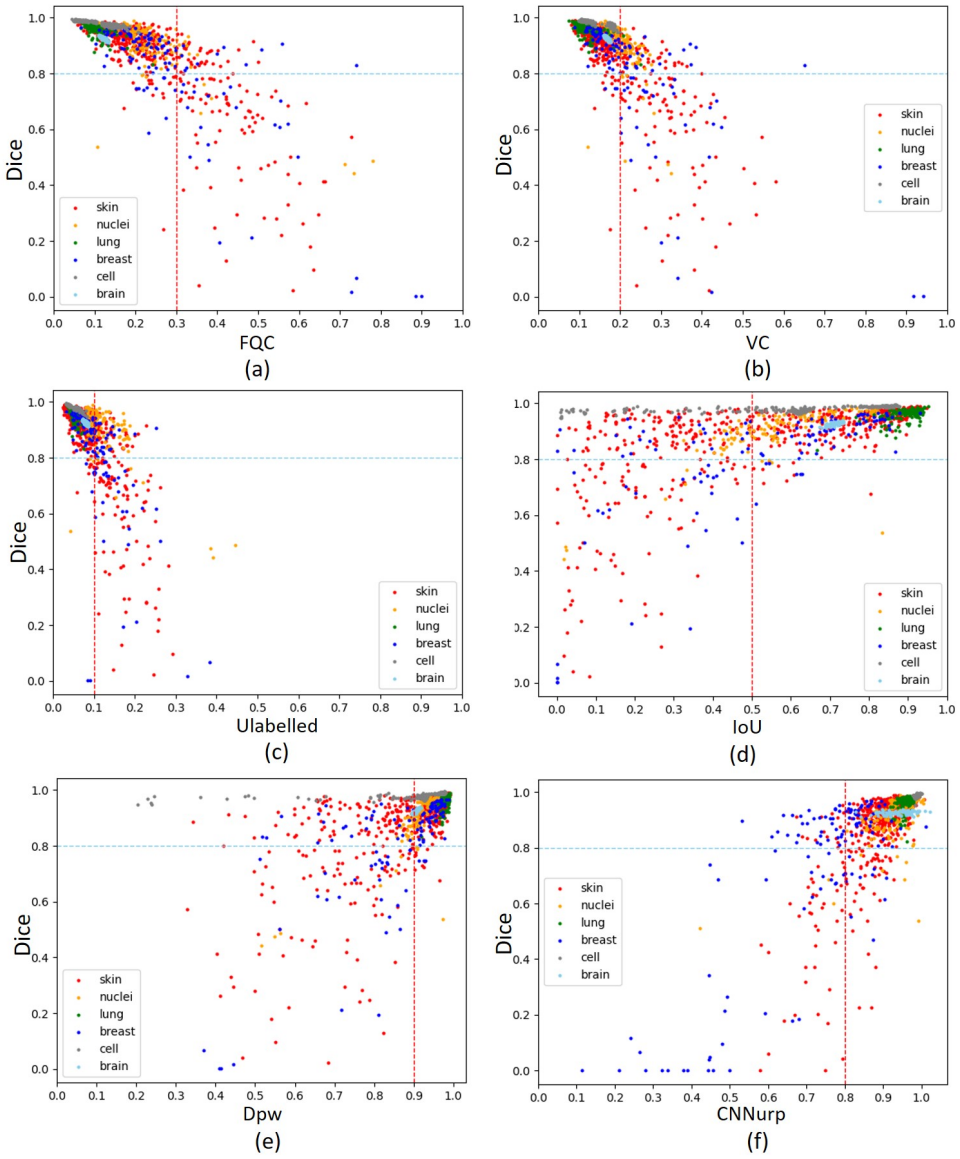


Figure 5.7: Scatter plots for the relationship between the uncertainty-based quality quantification methods and the segmentation quality measurement (the Dice coefficient). Each points refers to one test image for the given five datasets. The sky-blue dots line refers to the threshold (Dice = 0.8) of truly good or poor segmentation images. The red dots line means the threshold for the six uncertainty-based quality quantification methods to classify the good or poor segmentation images. Note that as FQC, VC and Ulabelled have a negative relationship with the Dice coefficient, the left part of the red dots line is the predicted good segmentation images and the right part of the red dots line is the predicted poor segmentation images. Whereas IoU, Dpw and CNNurp have a positive relationship with the Dice coefficient, thus the right part of the red dots line is the predicted good segmentation images and the left part of the red dots line is the predicted poor segmentation images.

have a positive relationship with the Dice coefficient. Furthermore, the outliers that impact the linear relationship in FQC are fewer than those in other quality quantification methods, which is consistent with the conclusion drawn based on the PCC value in Table 1.

To further study whether FQC has a better performance than other uncertainty-based quality quantification methods with statistical significance, Friedman test [112] and Bonferroni-Dunn post-hoc test [113] are applied on the five datasets. Given M quality quantification algorithms and N datasets, r_i^k refers to the rank of the k^{th} algorithm on the i^{th} dataset. The average rank for each algorithm is $R_k = \frac{1}{N}\sum_{i=1}^{N} r_i^k$. The null-hypothesis is that all the algorithms share an equal rank. Then Friedman test is applied to check whether we can reject the null-hypothesis and whether all the quality quantification algorithms are significantly different. Based on the average rank of each algorithm given in Table 5.1, the Friedman statistic value is calculated by $F = \frac{(N-1)\chi^2}{N(M-1)-\chi^2}$, where $N = 6$, $M = 8$, $\chi^2 = \frac{12N}{M(M+1)}(\sum_{k=1}^{M}$
$R_k^2 - \frac{M(M+1)^2}{4})$. F follows the F-distribution with $M-1 = 7$ and $(M-1)(N-1) = 35$ degrees of freedom and the related critical value is 2.285 given the significance level $\alpha = 0.05$. Thus, the null-hypothesis can be rejected due to the fact $F = 6.699 > 2.285$, which means that there are statistic difference between all the eight quality quantification algorithms.

Then the Bonferroni-Dunn post-hoc test is adopted to explore whether the FQC method is better than VC, Ulabelled, IoU, Dpw and CNNurp. If the difference between the average ranks of two algorithms is greater than the critical distance (CD), we could conclude that these two algorithms have significantly different performances. The CD is calculated by the formula $CD = q_\alpha\sqrt{\frac{T(T+1)}{6Z}} = 3.464$, where T is the number of algorithms to be compared, and Z is the number of datasets. q_α is the critical value given the significance level α [114]. In conclusion, IOU ($6 - 2.2 > CD$), Dpw ($5.8 - 2.2 > CD$) and CNNurp ($7.2 -$

$2.2 > CD$) have a significant difference with FQC. Therefore, FQC performed significantly better than other methods which are all using the averaged fusion method.

5.3.3 Application of Uncertainty for Quality Quantification

The quality quantification algorithm aims to generate a value to infer the image segmentation quality. Given a threshold, if the generated value is greater/less than the threshold, the segmentation result is regarded as good/poor quality, which can be treated as a binary classifier. If one quality quantification algorithm has good classification capability, it has a higher feasibility to be applied in practice. Thus, the following experiment explores how the optimal threshold can be defined and compares the ability of different quality quantification algorithms to classify the segmentation images into good/poor quality.

Figure 5.8 shows the process of searching thresholds. Taking the proposed FQC algorithm as an example, the points in Figure 5.8 refers to the test images for all five datasets; yellow points indicate the truly good-quality segmentation images with the Dice ≥ 0.8 and their corresponding labels are 1; green points indicate the truly poor-quality segmentation images with the Dice < 0.8 and their corresponding labels are 0. The red line indicates the threshold and can be considered as a binary classifier. As the output of FQC has a negative relationship with the Dice coefficient, the left part of the red line is the predicted 1 and the right part is the predicted 0. Thus, as shown in Figure 5.8, ① is the true positive (TP), ② is the False Positive (FP), ③ is the False Negative (FN), and ④ is the True Negative (TN). By moving the red line, different binary classifiers are created to generate numerous (FPR, TPR) pairs, where $\text{FPR} = \frac{\text{FP}}{\text{FP+TN}}$, and $\text{TPR} = \frac{\text{TP}}{\text{TP+FN}}$.

Then the receiver operating curves (ROC) of all compared methods are plotted

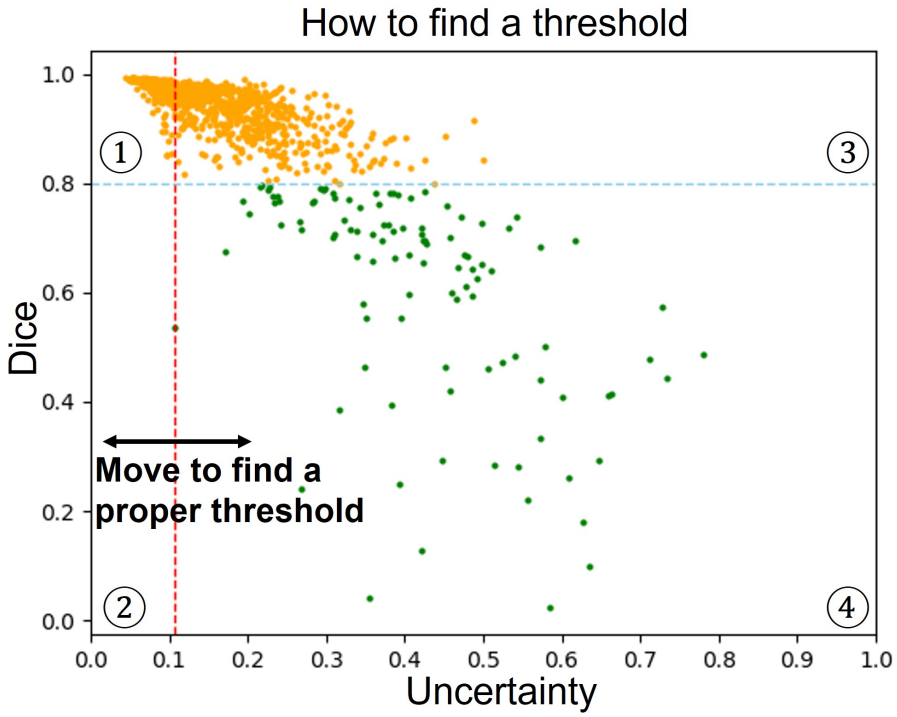


Figure 5.8: The process of searching the optimal threshold. Yellow points indicate the truly good quality segmentation images and green points indicate the truly poor quality segmentation images. The red line indicates the threshold and can be considered as a binary classifier. The optimal threshold represents the classifier has the best performance.

in Figure 5.9 based on the (FPR, TPR) pairs. Note that IoU, Dpw and CNNurp have a positive relationship with the Dice coefficient. Therefore, for IoU, Dpw and CNNurp, the left part of the red line is the predicted 0 and the right part is the predicted 1, which is opposite to FQC, VC and Ulabelled. By using the same approach mentioned above, the ROCs of other quality quantification methods are obtained and shown in Figure 5.9. The area under the ROC, namely AUC, is a criterion to measure the classification capability of the classifier. The AUCs for the proposed FQCfuzzy and FQC methods are significantly larger than the other five quality quantification algorithms. The optimal threshold for each quality quantification method is the threshold corresponding to the $\max(\text{TPR} - \text{FPR})$ in ROC, which is the red line shown in Figure 5.7.

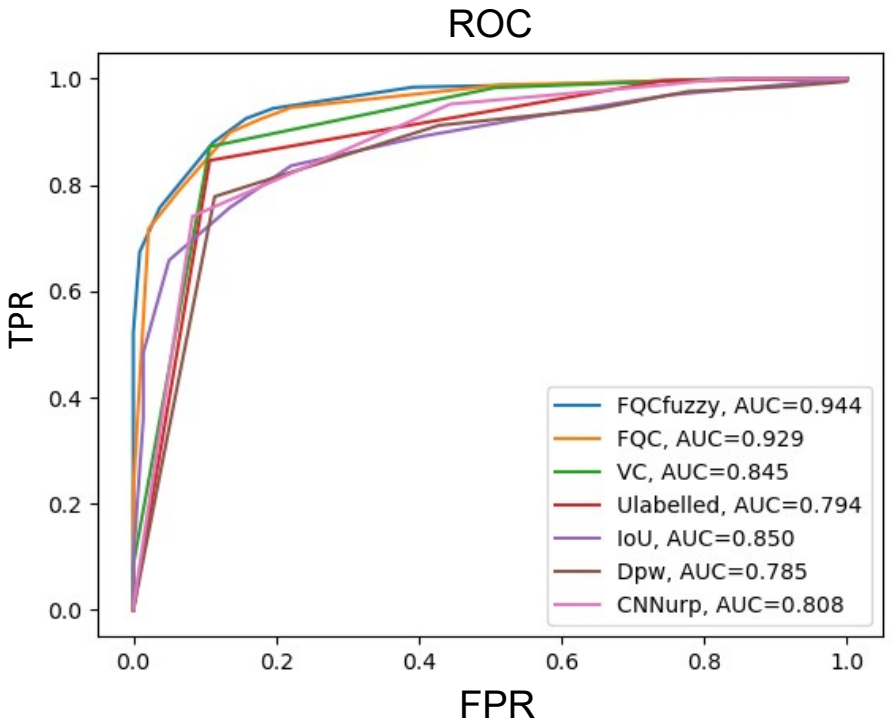


Figure 5.9: The ROC curves for FQCfuzzy, FQC, VC, Ulabelled, IoU, Dpw and CNNurp. The area under the ROC curve namely AUC is a a criterion to measure the classification capability of the classifier.

5.4 Discussion

Segmentation uncertainty consists of data uncertainty and model uncertainty. Thus, it is sensible that uncertainty-based quality quantification algorithms adopt the TTA and MCdropout simultaneously in order to improve the accuracy in uncertainty computation (see the results for FQC (TTA + MCdropout) and Fdata (TTA only) in Table 5.1).

Table 5.1 also shows that the proposed algorithm with the fuzzy image fusion method (FQCfuzzy) performs better than that with the average image fusion method (FQC) on skin, nuclei, breast and cell datasets, while performs worse on the lung dataset. This performance variation in datasets is due to the fact that lung has a more consistent boundary shape across different subjects, whereas skin, nuclei, breast and cell datasets have various and compli-

Table 5.2: The average computational time of FQC and FQCfuzzy for one test image on the five given datasets

Method	Skin	Nuclei	Lung	Breast	Cell
FQC	2.88s	2.86s	3.20s	2.97s	2.73s
FQCfuzzy	24.20s	24.65s	24.80s	24.34s	24.81s

cated object boundaries. This indicates that FQCfuzzy is better at handling 2D datasets with complicated segmentation boundaries. To further explore the reasons, the calculation procedure of the fuzzy image fusion method is investigated. It is noted that uncertainty generally happens at segmentation boundaries. If the given image has complex boundaries, the boundary pixel values in the same location of N predicted images generated by TTA and MCdropout are inconsistent and are likely to include outliers. The fuzzy fusion method based on Algorithm 5.1 assigns high weights to the main pixel values and low weights to the outliers, improving the accuracy of fusion results by eliminating the influence of outliers. For example, given a boundary pixel value set $(0.01, 0.91, 0.94, 0.94, 0.94, 0.99, 0.98, 0.01)$, the fuzzy fusion method assigns a weight set $(0, 1, 1, 1, 1, 1, 1, 0)$ based on the membership functions shown in Figure 5.3 to the given pixel set and the fusion result is 0.95. However, the average fusion method is not robust to outliers since it treats each pixel value as having the same weight. Thus, the fusion result with average fusion method is 0.715, which is inaccurate and results in errors in the final uncertainty computation. However, as FQCfuzzy is a pixel-wise fusion method, and needs to use three membership functions to determine the weights for each pixel, it is more time-consuming in comparison with FQC, as illustrated in Table 5.2.

Next the advantages and limitations of these quality quantification methods are discussed. Compared to FQC, Dpw and IOU rely heavily on each predicted image. Based on the formulas of Dpw and IOU given in [47] and [48], if one of the N predicted images has a poor-quality segmentation result, the Dpw and

IoU are likely to be influenced considerably. Table 5.3 presents the effects of poor-quality segmentation results for IoU, Dpw and FQC. Given one test image from the skin cancer dataset, 24 predicted images are generated. If there is one poor-quality predicted image, the value of IoU falls dramatically. With an increasing number of poor-quality predicted images, the value of Dpw also decreases significantly. However, for the proposed FQC method, the value varies only slightly. Moreover, the Dice coefficient refers to the real segmentation performance of the given test image calculated by the average of 24 predicted images and the ground truth image. The poor-quality predicted images have little effect on the Dice coefficient value. From Figure 5.7, Dpw and IoU have a positive relationship with the Dice coefficient and FQC has a negative relationship with the Dice coefficient. With the increase of the Dice coefficient value, the IOU and Dpw should also increase but they fall significantly due to the impact of the poor-quality predicted images. In contrast, FQC has a consistent trend with the Dice coefficient due to the fact that FQC has the ability to handle the impact of the poor-quality predicted images. Therefore, FQC is more stable and robust than IoU and Dpw.

Like FQC, the VC method takes the segmentation region variation of N predicted images into consideration. However, FQC divides the segmentation region into K groups based on the distance transform algorithm, which helps distinguish between areas with high uncertainty and low uncertainty. As the inaccurate segmented pixels often occur in the area with high uncertainty, focusing on the area with high uncertainty could potentially improve the performance. In contrast, VC treats the low uncertainty area and the high uncertainty area equally which is not conducive to accurately capturing the mis-classified parts especially for datasets with multiple segmentation regions. Thus, FQC performs better than the VC on most datasets shown in Table 5.1.

The Ulabelled method takes the pixel-wise variation of N predicted images into

Table 5.3: The influence of poor-quality segmentation results on IoU, Dpw and FQC methods

Method	No poor-quality predicted image	One poor-quality predicted image	Three poor-quality predicted images
IoU	0.932	0.141	0.140
Dpw	0.989	0.928	0.872
Fuzzy	0.143	0.137	0.131
Dice	0.973	0.975	0.979

consideration, which ignores the influence of the neighbouring pixels. In contrast, although FQC is a region-wise quality quantification method in theory, it is capable of balancing pixel-wise variation and region-wise variation due to the fact that it divides the segmentation region into numerous groups. Hence, compared to Ulabelled, FQC performs better on almost all datasets except the cell dataset. The target objects in the cell dataset are all very small, so Ulabelled's pixel-wise based method is able to perform well.

The CNNurp method trains a Resnet regression model to predict the Dice coefficient based on the uncertainty map, the predicted image and the raw image. On the one hand, training a deep learning model is time-consuming compared with FQC. On the other hand, medical image datasets are generally not big datasets, which may cause model over-fitting. Therefore, the CNNurp does not have a better performance on six datasets in comparison to the proposed FQC method.

It is worth noting that the data augmentation operators of the training process may have an impact on the uncertainty calculation in the proposed quality quantification method. For example, if the training and testing processes use similar data augmentation operators such as scaling, rotation, and flipping, it is expected to have a low uncertainty score of the output. However, when the data augmentation transformation is different for training and testing images (e.g. Training process: scaling, rotation, flipping. Testing process: rotation, color augmentation, brightness), the uncertainty would be higher. Therefore, the proposed

method is based on the assumption that the quality quantification algorithm is independent of the model training process and they do not share a similar data augmentation process. We only used data augmentation during the test time.

In addition, as mentioned in Section 5.2.3, the segmentation region is divided into K groups and each group has numerous pixels even from a single predicted image. Those pixels in the K groups naturally form a type-2 fuzzy set. Then the type-2 fuzzy set is directly converted to a type-1 fuzzy set for uncertainty estimation in the proposed method. However, it is commonly acknowledged that type-2 fuzzy sets are better at representing uncertainty [115]. In our study, it is an initial step in exploring the use of a type-2 fuzzy set in deriving the segmentation quality. More sophisticated techniques can be explored in future work.

5.5 Summary

In this chapter, a novel fuzzy-uncertainty-based quality quantification method is proposed to address objective 3. This algorithm consists of three parts: adopting TTA and MCdropout to capture the data uncertainty and model uncertainty, using a fuzzy set to describe the captured uncertainty, and utilizing the fuzziness to calculate the image-level uncertainty value. Extensive experiments using six medical image segmentation applications on the detection of skin lesion, nuclei, lung, breast, cell, and brain are conducted to evaluate the proposed algorithm. The experimental results show that the estimated image-level uncertainties using the proposed method have strong correlations with the segmentation qualities measured by the Dice coefficient. Although the fuzzy image fusion method outperforms the average fusion method in the proposed algorithm, its time complexity is very high and therefore may not be the best option depending on the

application. Compared to other state-of-the-art quality quantification methods (VC, Ulabelled, IoU, Dpw, and CNNurp), the proposed FQC algorithm has a better ability to assess the segmentation quality and to classify the good/poor segmentation images.

In this chapter, the proposed fuzzy-uncertainty-based quality quantification algorithm has satisfying performance on six public datasets. To further investigate the practical application of fuzzy uncertainty in a clinical setting, the next chapter will include a real-world medical case study that involves quantitative and qualitative analyses.

Chapter 6

Real-world Medical Case Study

Last Chapter, a fuzzy-uncertainty-based quality quantification algorithm is proposed to indicate the success/failure or the level of trustworthiness of the segmentation results without access to the ground truth images. In fact, the proposed quality quantification algorithm is significantly useful in practical applications, especially in clinical settings. To explore how to apply quality quantification in clinical settings, this chapter goes beyond public datasets and delves into a real-world case study involving cardiac MRI segmentation. Moreover, as clinicians also provide the level of uncertainty to measure their confidence when annotating to generate ground truth images (human-based uncertainty), the correlation between human-based uncertainty and AI-based uncertainty (calculated by the proposed quality quantification algorithm) is investigated. In this chapter, by addressing objective 4, quantitative and qualitative analyses are implemented simultaneously to obtain experimental results based on the dataset and feedback from the clinicians, as presented in my study [116].

In Section 6.1, background information on cardiac magnetic resonance imaging (CMR) is provided. Section 6.2 details the dataset, segmentation model, the proposed fuzzy-uncertainty-based quality quantification algorithm, and defini-

tions of five different kinds of AI-based uncertainties. Section 6.3 investigates the performance of AI-based uncertainty for the cardiac MRI dataset from four aspects. The comparison of AI-based uncertainty and human-based uncertainty is presented in Section 6.4. Section 6.5 conducts a qualitative study with clinicians to obtain their feedback. Section 6.6 provides discussions about the differences between AI-based uncertainty and human-based uncertainty and their practical applications in clinical settings. Finally, the findings of this chapter are concluded and summarized in Section 6.7.

6.1 Background and Motivation

Recent research has shown that the size of ventricular scar has a strong relationship with the risk of ventricular arrhythmia episodes [117]. Thus it is crucial to correctly identify and quantify cardiac scar from medical images. Due to its ability to provide detailed and precise structure characterization, cardiac magnetic resonance imaging (MRI) is essential in not only diagnosing cardiac pathologies but also guiding appropriate treatment [118]. Cardiac MRI represents the gold standard for non-invasive cardiac structure characterization, detection of acute and chronic myocardial changes, and myocardial viability [119, 120].

Therefore, with its common use in clinical cardiology, cardiac MRI is ideal for quantitative scar measurements. Utilization of anatomical data provides complimentary benefits by allowing the segmentation of multiple structures. Classical image automatic segmentation technologies consist of edge detection [9], threshold [7], region growing [8], and clustering [35]. These methods are not robust and have low segmentation accuracy when the boundary of the original image is complicated and overlapped. To mitigate these problems, end-to-

end convolutional neural networks (CNNs)-based semantic segmentation techniques have gradually become the mainstream image segmentation algorithms and achieved outstanding performance [121, 122, 123]. However, one limitation of end-to-end CNNs-based segmentation models is that they only provide segmentation results without the corresponding information regarding the level of trustworthiness of the segmentation result, which hinders the widespread application of CNNs-based segmentation models in clinical practice.

It is crucial in clinical settings to provide accurate segmentation results as well as to inform the segmentation quality. As suggested by the experimental results on five 2D public datasets [91], the uncertainty derived from fuzzy-based quality quantification algorithm (AI-based uncertainty) [91] has a close linear relationship with the true segmentation quality. This indicates that AI-based uncertainty can serve as an effective metric to estimate the segmentation quality when ground truth images are unavailable. Thus, this chapter utilizes AI-based uncertainty to indicate the quality of the cardiac MRI segmentation model. Moreover, in clinical settings, the practical application of AI-based uncertainty is considerably complicated. A variety of uncertainties are introduced due the multiple slices and structures contained within each cardiac MRI study including class-level, slice-level, image-level, etc. Thus, this chapter also conducts qualitative analyses with clinicians to investigate all relevant uncertainties that may impact use clinically.

Furthermore, as some structures are complex and challenging to accurately segment for clinicians, uncertainty exists when performing annotations to generate ground truth images. This human-based uncertainty is graded by the clinicians. However, the relationship between AI-based uncertainty and human-based uncertainty is unknown and is investigated in this chapter.

6.2 Implementations Details

In this section, the cardiac MRI dataset, semantic segmentation model, and AI-based uncertainty algorithm are described in detail.

6.2.1 Materials

The cardiac dataset includes 483 multi-slice 2D late gadolinium-enhanced cardiac MRI (LGE-Cardiac MRI) from patients with ischemic cardiac disease. This cohort has been selected from clinically indicated routine scans (on Philips and Siemens scanners) undertaken at Nottingham University Hospitals NHS Trust and Leeds Institute of Cardiovascular and Metabolic Medicine, UK. Seven cardiac structures were manually labeled by Level 1 Society of CMR accredited cardiology operators. Manual segmentations were performed on the LGE-CMR short axis stack to delineate the left ventricle (LV) endocardium (LV-En), LV epicardium (LV-Ep), Scar, right ventricular (RV) endocardium (RV-En), RV epicardium (RV-Ep), papillary muscles (Pap) and aorta (Aor). During the annotation process, the clinical operator also utilized other MRI sequences (e.g. cine MRI, the long-axis stack MRI, and Phase-sensitive inversion recovery MRI) to assist the annotations. The final manual segmentation by our clinical expert was treated as the surrogate for "gold standard".

Due to the complexity of certain specific structures, accurately segmenting them is a challenge even for experts, resulting in the existence of uncertainty during the annotation process namely human-based uncertainty. Our human annotator also quantifies human-based uncertainty as the degree of uncertainty when performing manual delineation: 0 means high uncertainty, 1 refers to medium uncertainty, and 2 represents low uncertainty. This three-level uncertainty was recorded for each slice, which was easier to ensure consistency across slices and

subjects than a longer ordinal or continuous valued scale.

6.2.2 Segmentation Model

As one cardiac MRI consists of multi-slice 2D images, a 3D semantic segmentation model namely VNet [97] was applied to segment different structures in cardiac MRI dataset. The input and output image size for VNet is $16 \times 256 \times 256$, which means that all the datasets are reshaped to the same size. VNet has five layers and is constituted by an encoding process and a decoding process. During the encoding stage, the numbers of kernels for convolutional operators were 16 (first layer), 32 (second layer), 64 (third layer), 128 (fourth layer), and 256 (fifth layer), respectively. During the decoding stage, de-convolution operations with the kernel size $1 \times 2 \times 2$ were adopted to recover the feature map sizes.

For all the segmentation tasks, the datasets were divided into a training set (70%), a validation set (10%), and a testing set (20%). The parameters of VNet were initialized based on a uniform distribution [124]. VNet was trained with the combination of Cross-entropy Loss and Soft-Dice Loss. The weight for each loss was 0.5. During the training process, early stopping [125] was applied to avoid over-fitting issues. At the end of each epoch, the trained model was evaluated on the validation set. If the performance showed improvement, the model was saved as the temporary best model. If the performance on the validation set decreased for five consecutive epochs, the training process was terminated. If the entire training session did not trigger the early stopping mechanism, the model stopped training after 100 epochs. Adam optimization algorithm with an initial learning rate of 0.0001 was used to update the parameters for VNet. After obtaining the pre-trained segmentation models, various types of uncertainties were computed, as described in the next subsection.

6.2.3 AI-derived Uncertainty Generation

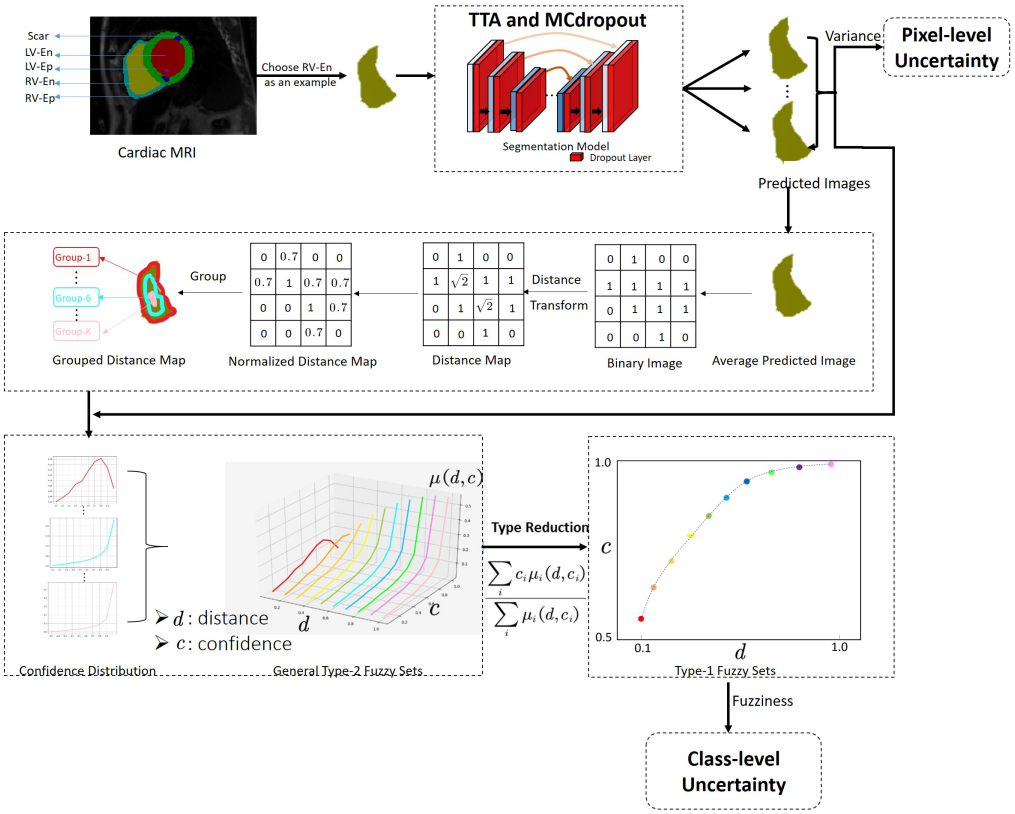


Figure 6.1: The Pipeline of the AI-based uncertainty algorithm

The pipeline of the AI-derived uncertainty algorithm [91] is shown in Figure 6.1. It is assumed that a DCNN-based segmentation model (VNet) is trained and capable of performing segmentation on a given input image. The AI-derived uncertainty algorithm works as a computational module in the inference procedure, which consists of the following steps: (1) Test-time augmentation (TTA) [99] and Monte Carlo Dropout (MCdropout) [100] are firstly applied to generate several predicted segmentation outputs for a given input image. (2) An average predicted segmentation mask is calculated based on these generated predictions. A distance map is generated from the average predicted segmentation mask using distance transform [89]. Each pixel value in the distance map represents the minimum distance to the boundary. The distances are then normalized and discretized into groups, resulting in a grouped distance map. (3) The predicted segmentation images are represented by a set of confidence distributions for all

the distance groups, which are then formalized as general type-2 fuzzy sets. (4) Type reduction is applied to convert the type-2 fuzzy sets to type-1 fuzzy sets. (5) Fuzziness measure is subsequently applied to the type-1 fuzzy sets to calculate an uncertainty value, which is used to quantify segmentation quality. The detailed process of each step is presented in my previous work [91].

The above AI-derived uncertainty approach is designed to measure the uncertainty in each class using fuzziness [91]. This allocates an individual uncertainty value to each class. Based on this method, when analyzing a multi-class cardiac MRI image, five types of uncertainties can be derived including pixel-level uncertainty, image-level uncertainty, class-level uncertainty, slice-level uncertainty, and slice-class-level uncertainty.

These uncertainties are explained as follows.

(1) Pixel-level uncertainty: after using the TTA and MCdropout, each pixel of the predicted image has several different values. Then each pixel's uncertainty can be calculated by variance [26]. Thus, pixel-level uncertainty can be visualized using a heat map with the same size as the raw image, which could assist clinicians in determining the regions that have high uncertainty and are possibly incorrectly predicted.

(2) Class-level uncertainty: for each individual class, the AI-derived uncertainty algorithm directly generates one uncertainty value to infer the level of uncertainty. Availability of this uncertainty could focus the clinician's attention to specific categories of interest.

(3) Image-level uncertainty: a signal uncertainty value is generated to measure the overall uncertainty of the image. The mean of class-level uncertainties is used to calculate the image-level uncertainty. This uncertainty score would identify overall poorly performing segmentation to clinicians

[126].

(4) Slice-level uncertainty: cardiac MRI scans consist of multi-slice 2D images. Thus for each slice, an uncertainty value is calculated to quantify its level of uncertainty. Like image-level uncertainty, slice-level uncertainty may help clinicians detect poor segmentation results at the slice level.

(5) Slice-class-level uncertainty: considering that each slice can be treated as a 2D image with multiple classes, the uncertainty of each class within a given slice is quantified using a signal value. This kind of uncertainty provides more detailed uncertainty information for clinicians and is highly beneficial when clinicians intend to investigate the class-level segmentation quality of each slice.

6.3 Uncertainty-based Quality Quantification

quality quantification serves as an auxiliary algorithm designed to assess the segmentation quality in the absence of ground truth images. My work [91] proposed fuzzy-based uncertainty to conduct quality quantification. Herein the proposed quality quantification algorithm was firstly applied to the cardiac MRI dataset, and its effectiveness was assessed.

3D VNet is utilized for segmenting the cardiac MRI dataset. The true segmentation quality is measured by the Dice coefficient (DC). DC is calculated by the comparison of the predicted segmentation mask with the ground truth mask. As one cardiac MRI is a multi-class image, DC can be calculated in four levels. 1) Image-level DC: it measures the overall segmentation quality of the given cardiac MRI image. 2) Class-level DC: it estimates each structure's segmentation quality. 3) Slice-level DC: since a cardiac MRI image comprises multiple 2D slices, the slice-level DC is utilized to evaluate the segmentation quality of

each individual slice. 4) Slice-class-level DC: for each 2D slice, it also includes multiple classes and each class has a DC to measure its segmentation quality.

As mentioned in Section 6.2.3, the proposed uncertainty quantification algorithm was used to measure AI-based uncertainty from image-level, class-level, slice-level, and slice-class-level perspectives. Thus, the correlation between the AI-based uncertainty and DC is explored. The Pearson correlation coefficient (PE) measures the linear correlation of DC and the AI-based uncertainty, and the value is between -1 and 1. When the value is close to 1 or -1, it means the given two variables have a strong positive or a strong negative relationship, respectively. The closer a correlation value is to 0, the less correlation can be found between the two variables. The formula of the Pearson correlation coefficient is given as

$$\rho_{X,Y} = \frac{\mathbb{E}[XY] - \mathbb{E}[X]\,\mathbb{E}[Y]}{\sqrt{\mathbb{E}[X^2] - (\mathbb{E}[X])^2}\sqrt{\mathbb{E}[Y^2] - (\mathbb{E}[Y])^2}} \tag{6.1}$$

where X and Y refer to uncertainty values and DC values respectively, $\mathbb{E}$ means the expectation.

In the experiments, the testing set is used to test the performance of VNet segmentation model (DC) and the proposed quality quantification algorithm (PE). The corresponding experimental results are shown in Table 6.1. It can seen that the PE is high despite the DC value being low, e.g. DC for scar is only 0.36 but PE is -0.76. This suggests that AI-based uncertainty is a good segmentation quality indicator. Figure 6.2 visualizes the relationship between AI-based uncertainty and DC at image-level, class-level, slice-level, and slice-class-level. Note that when all classes are integrated into one figure (Figure 6.2(b) and Figure 6.2(d)), the mean of DC (y-axis) and the PE between DC and AI-based uncertainty are calculated and recorded as 'Overall' in Table 6.1. From Fig-

ure 6.2(b) and Figure 6.2(d), it can be seen that DC and AI-based uncertainty have a strong relationship (PE=-0.922/-0.928) when all classes are considered. Thus PE values of "Overall" are utilized to represent the quality quantification performance of class-level and slice-class-level. Moreover, the class-level (PE=-0.922) and slice-class-level (PE=-0.928) have higher Pearson values compared to the image-level (PE=-0.838) and slice-level (PE=-0.798). It suggests that class-level including slice-class-level uncertainty has a strong linear negative relationship with the true segmentation quality (measured by DC). When there are no ground truth images, the class-level uncertainty could be used to infer segmentation quality effectively [91]. To determine whether the class-level uncertainty is useful in clinical settings, further investigation is conducted in collaboration with clinicians.

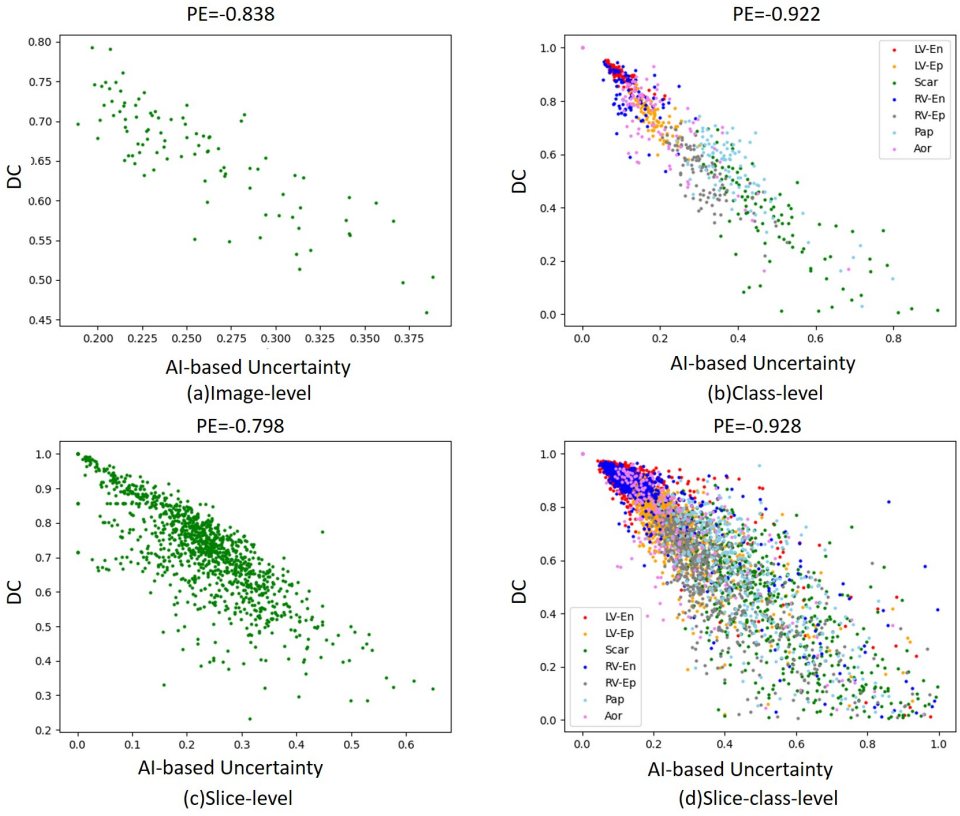


Figure 6.2: Scatter plot for the relationship between the DC and AI-based uncertainty. PE represents the Pearson correlation between AI-based uncertainty and DC.

Table 6.1: The true segmentation quality of cardiac MRI dataset and the Pearson correlation coefficient (PE) between DC and AI-based Uncertainty

Perspective	Class	DC	PE
Image-level	-	0.658	-0.838
Class-level	LV-En	0.902	-0.855
	LV-Ep	0.741	-0.784
	Scar	0.364	-0.762
	RV-En	0.827	-0.529
	RV-Ep	0.528	-0.795
	Pap	0.527	-0.816
	Aor	0.732	-0.825
	Overall	0.663	-0.922
Slice-level	-	0.724	-0.798
Slice-class-level	LV-En	0.905	-0.883
	LV-Ep	0.742	-0.837
	Scar	0.661	-0.933
	RV-En	0.890	-0.914
	RV-Ep	0.671	-0.919
	Pap	0.823	-0.944
	Aor	0.947	-0.916
	Overall	0.811	-0.928

6.4 Comparison of AI-based Uncertainty and Human-based Uncertainty

As described in Section 6.2.1, when experts conducted manual annotation, they also quantified uncertainty using 0, 1, and 2 at slice level. The uncertainty

quantified by experts is referred to as human-based uncertainty in the remaining parts of this chapter. To fairly investigate the difference between human-based uncertainty and AI-based uncertainty (calculated by the algorithm described in Section 3.3), slice-level and slice-class-level AI-based uncertainties are used in the following experiments.

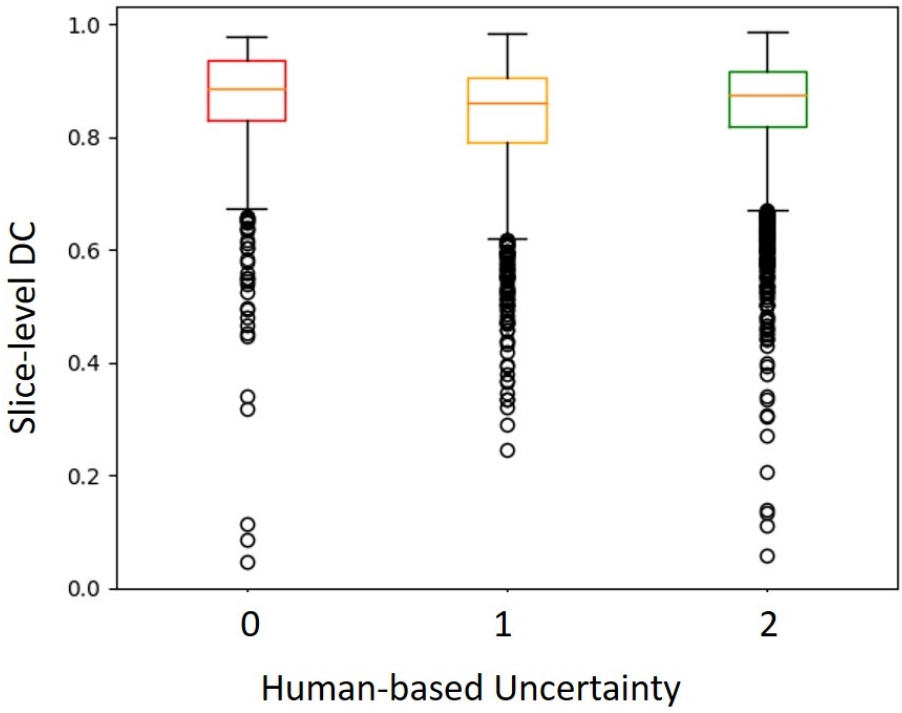


Figure 6.3: The box plot of human-based uncertainty and the true slice-level segmentation quality (measured by slice-level DC) .

Table 6.2: The statistic values for each box of slice-level DC; $*$ means box-1 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05

Human-based Uncertianty	Slice-level DC		
	Mean	std	median
0	0.839	0.151	0.884
1	0.834	0.108	0.861
2$*$	0.849	0.104	0.874

Figure 6.3 and Table 6.2 show the relationship between the true segmentation quality measured by DC and human-based uncertainty. Note that DC is calculated slice by slice. As can be observed from Figure 6.3 and Table 6.2,

when clinical experts are uncertain about their annotation (human-based uncertainty=0), the AI-based segmentation model can still potentially perform reasonably well with the segmentation quality sometimes even outperforming when the uncertainty level is 1. This suggests a poor correlation between human-based uncertainty and DC. In other words, high uncertainty labeled by humans does not necessarily lead to low segmentation quality using AI-based segmentation models.

Next, the relationship between human-based and AI-based uncertainties at slice level is investigated.

Figure 6.4 shows the relationship between AI-based uncertainty and human-based uncertainty at slice level. Then statistical results of AI-based uncertainty at each level of human-based uncertainty are presented in Table 6.3. From the box plot and statistical values, it can be observed that when human-based uncertainty is equal to 0 or 1, there is no statistical difference to the corresponding AI-based uncertainties. When human-based uncertainty is equal to 2, AI-based uncertainty values are lower compared to the ones at the other two levels of human-based uncertainties. The independent samples' T-test further verifies that box-0 or box-1 has a statistically significant difference from box-2 (all p values < 0.05). This suggests that when the clinician has a lower level of uncertainty (level 2), the AI-based uncertainty value is also lower than other levels.

Table 6.3: The statistic values for each box of slice-level AI-based uncertainty; $\sharp$ means box-0 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05; $*$ means box-1 and box-2 are significantly different with P value <0.05

Human-based Uncertainty	Slice-level AI-based Uncertainty		
	Mean	std	median
0	0.262	0.124	0.252
1	0.261	0.098	0.251
$2\sharp*$	0.234	0.093	0.223

As the cardiac MRI dataset consists of 7 classes: LV-En, LV-Ep, Scar, RV-En,

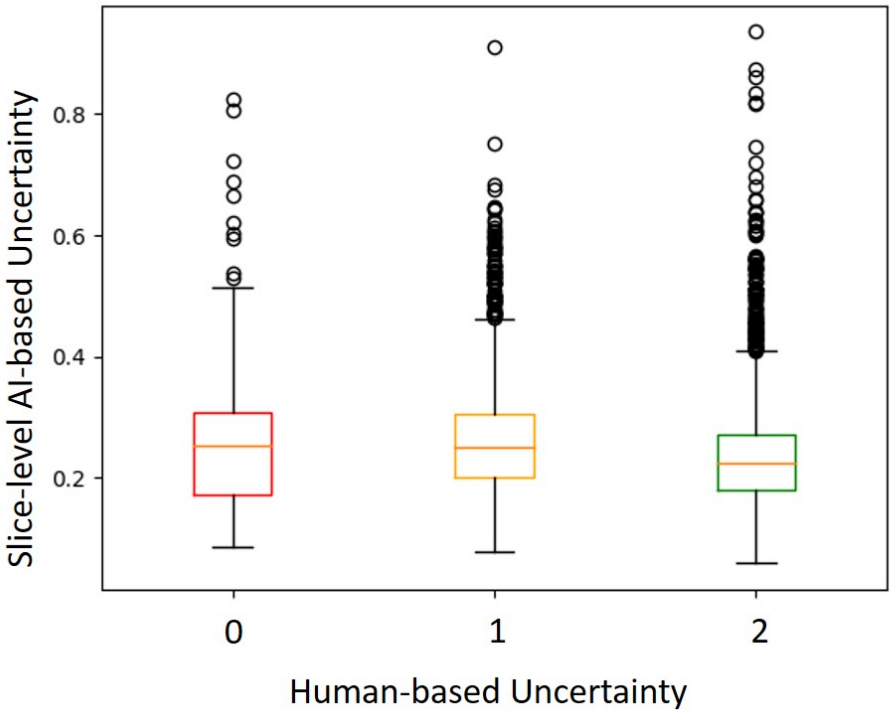


Figure 6.4: The box plot for AI-based uncertainty and Human-based uncertainty on the slice level.

RV-Ep, Pap, and Aor, each slice also consists of these classes, and each class in the same slice shares the same human-based uncertainty. Hence, the next step is to analyze the correlation between human-based uncertainty and AI-based uncertainty at the slice-class level. In Figure 6.5 and Table 6.4, for LV-En, Scar, and RV-En, box-0 and box-1 show no statistical difference while box-0/box-1 and box-2 are statistically different. For LV-Ep, RV-EP, and Pap classes, each box demonstrates statistically significant differences when compared to one another. This indicates that AI-based uncertainty has a consistent trend with human-based uncertainty. Nevertheless, the correlation between human-based uncertainty and AI-based uncertainty is not clear for the Aor class.

Furthermore, by observing some AI-based results, one conjecture is that the AI-based uncertainty has a potentially closer relationship with the structure's size than human-based uncertainty. Then for each slice, the proportion of the

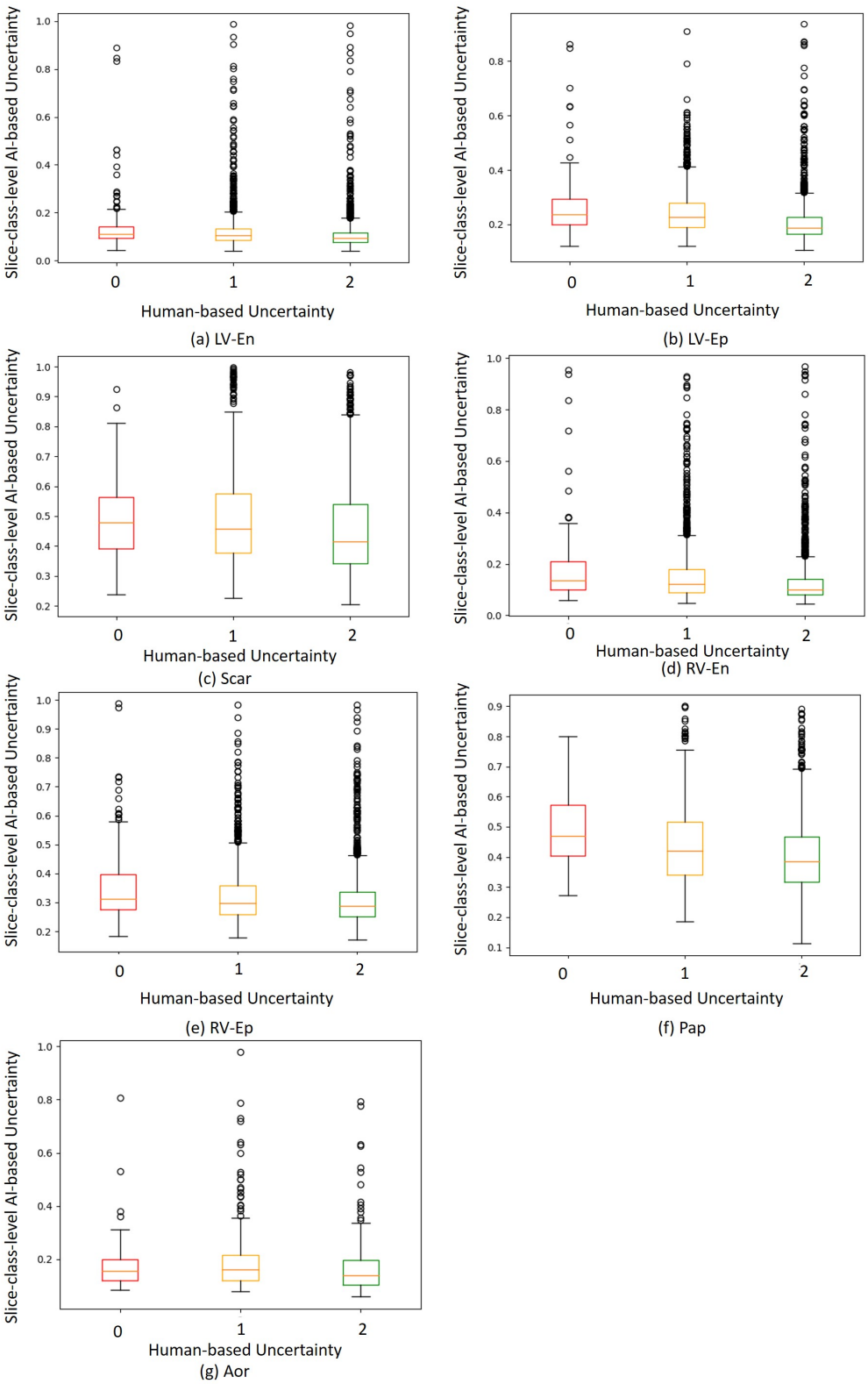


Figure 6.5: The box plot of AI-based Uncertainty and Human-based Uncertainty for each class.

structures for the whole slice was calculated. AI-based uncertainty was divided into 0, 1 and 2 to mirror human-based uncertainty for balanced comparison. First, the maximum (max) and minimum (min) values of the AI-based uncertainty were obtained. Then if the AI-based uncertainty value was greater than $2 \times (\max - \min)/3$, it was represented as 0 suggesting high uncertainty. If the AI-based uncertainty value was less than $(\max - \min)/3$, it was represented as 2 suggesting low uncertainty. If the AI-based uncertainty value was in the range of $[(\max - \min)/3, 2 \times (\max - \min)/3]$, it was represented as 1 suggesting medium uncertainty.

Figure 6.6 shows the relationships between slice-level size and uncertainty (human-based and AI-based) using box and scatter plots. From Figure 6.6 (a)-(b), despite the increase in structure size, both the human-based uncertainty and AI-based uncertainty overall correspond in certainty. The discrepancy between the three boxes is significantly greater for AI-based uncertainty in comparison to human-based uncertainty. Figure 6.6 (b) clearly shows that AI models exhibit high uncertainty when the structure size is relatively small. Therefore, AI-based uncertainty has a stronger relationship with structure size compared to human-based uncertainty.

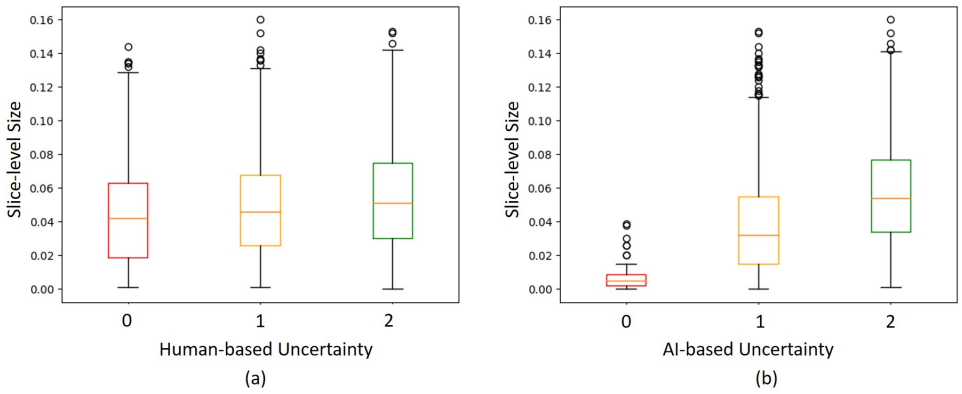


Figure 6.6: (a) is the box plot for the relationship between human-based uncertainty and slice-level structure' size. (b) is the box plot for the relationship between AI-based uncertainty and structure' size at the slice level.

Table 6.4: The statistic values for each box of slice-class-level AI-based uncertainty; ♯ means box-0 and box-2 are significantly different measured by the independent samples' T-test with P value <0.05; $*$ means box-1 and box-2 are significantly different with P value <0.05; ‡ means box-0 and box-1 are significantly different with P value <0.05

Class	Human-based Uncertainty	Slice-class-level AI-based Uncertainty		
		Mean	std	median
LV-En	0	0.139	0.111	0.112
	1	0.126	0.087	0.105
	2♯$*$	0.108	0.069	0.094
LV-Ep	0	0.262	0.106	0.236
	1‡	0.245	0.082	0.227
	2♯$*$	0.206	0.075	0.188
Scar	0	0.487	0.137	0.480
	1	0.488	0.154	0.457
	2♯$*$	0.454	0.156	0.414
RV-En	0	0.189	0.163	0.136
	1	0.165	0.134	0.121
	2♯$*$	0.131	0.101	0.100
RV-Ep	0	0.363	0.144	0.313
	1‡	0.326	0.106	0.298
	2♯$*$	0.309	0.097	0.287
Pap	0	0.493	0.132	0.471
	1‡	0.442	0.136	0.419
	2♯$*$	0.405	0.124	0.384
Aor	0	0.181	0.101	0.157
	1	0.189	0.114	0.160
	2$*$	0.163	0.091	0.139

6.5 Qualitative Analysis

In this section, a qualitative analysis is conducted to explore the clinical application of AI-based uncertainty and the difference between AI-based uncertainty and human-based uncertainty with two clinicians. The two clinicians are consultant cardiologists with a focus on cardiac rhythm management at Nottingham University Hospitals NHS Trust. They provided the human-based uncertainty and annotated the cardiac MRI raw images used in this project.

6.5.1 Clinical Application of AI-derived Uncertainty

Section 6.2.3 described various uncertainties: pixel-level uncertainty, image-level uncertainty, class-level uncertainty, slice-level uncertainty, and slice-class-level uncertainty. To explore which uncertainty is useful and how these uncertainties are used clinically, the meanings and definitions of these uncertainties were explained to clinicians using the examples in Appendix, thereby obtaining their feedback and insight.

Based on the feedback, both clinicians agreed that pixel-level uncertainty provides a quick overview of areas where the segmentation model finds challenging and could be used to highlight areas for segmentation improvement. Additionally, as pixel-level uncertainty is presented as a map rather than a numerical value, it is easily and rapidly comprehended. The potential clinical applications of pixel-level uncertainty include 1) pre-procedural review by non-imaging specialists for a general understanding of the reliability of segmentation; 2) second check for imaging cardiologists.

As discussed above, image-level uncertainty is the average of all class-level uncertainties. According to the clinicians, in clinical settings, not all categories

are important depending on the question. For example, clinicians may pay more attention to the scar class for diagnostics, risk prediction or procedural guidance. Therefore, class-level uncertainty provides more detail and is potentially more clinically valuable in comparison to image-level uncertainty. Furthermore, the potential clinical application areas of class-level uncertainty are broad including image analysis (quantification of classes) and risk prediction (identifying the classes with poor segmentation quality).

Slice-level and slice-class-level uncertainties are calculated slice by slice. Based on the clinicians' feedback, the slice-level and slice-class-level uncertainties are only useful if image quality is not homogeneous for all slices, i.e., poor apical or basal slices that would cause significant deterioration. One potential application is to further interrogate the data to see if certain slices could be assessed for exclusion from segmentation.

Overall, both clinicians agreed that pixel-level uncertainty and class-level uncertainty are potentially more useful and significant in clinical settings compared to other uncertainties.

6.5.2 Generation of Human-based Uncertainty

To conduct a qualitative analysis of the difference between AI-based uncertainty and human-based uncertainty, some representative cases are visualized to discuss with the clinicians (shown in Figure 6.7–6.9). It should be noted that slices with low AI-based uncertainty but a human-based uncertainty of 0/1, or slices with high AI-based uncertainty but a human-based uncertainty of 2, are used as the case selection criteria.

In Figure 6.7 and 6.8, each slice only has one structure, and the shape of this structure is easily distinguishable and straightforward to be segmented by the

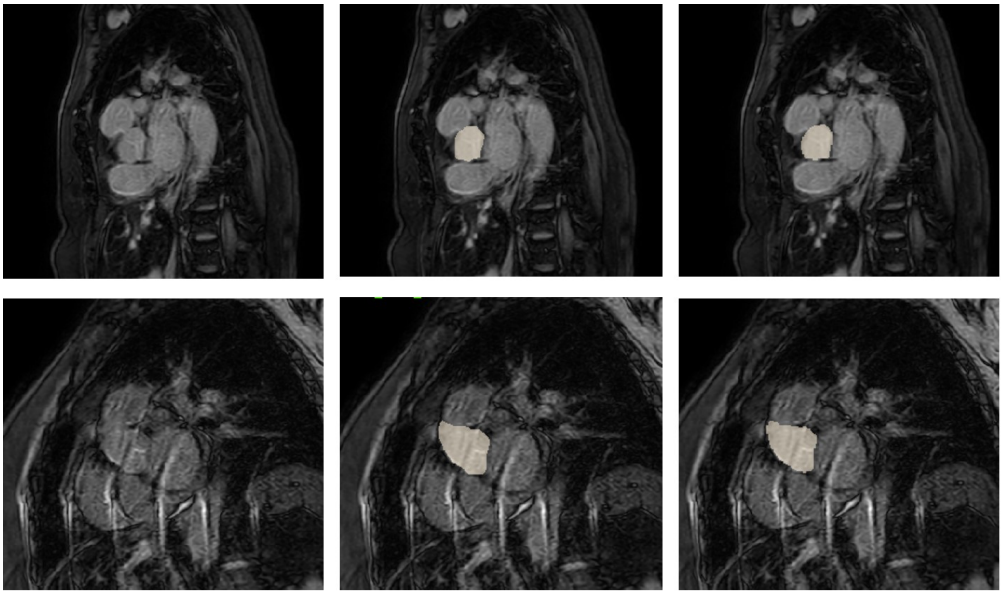


Figure 6.7: Two cases with low AI-based uncertainty (0.085 and 0.095) and high human-based uncertainty (level 0). The tissue is the Aor.

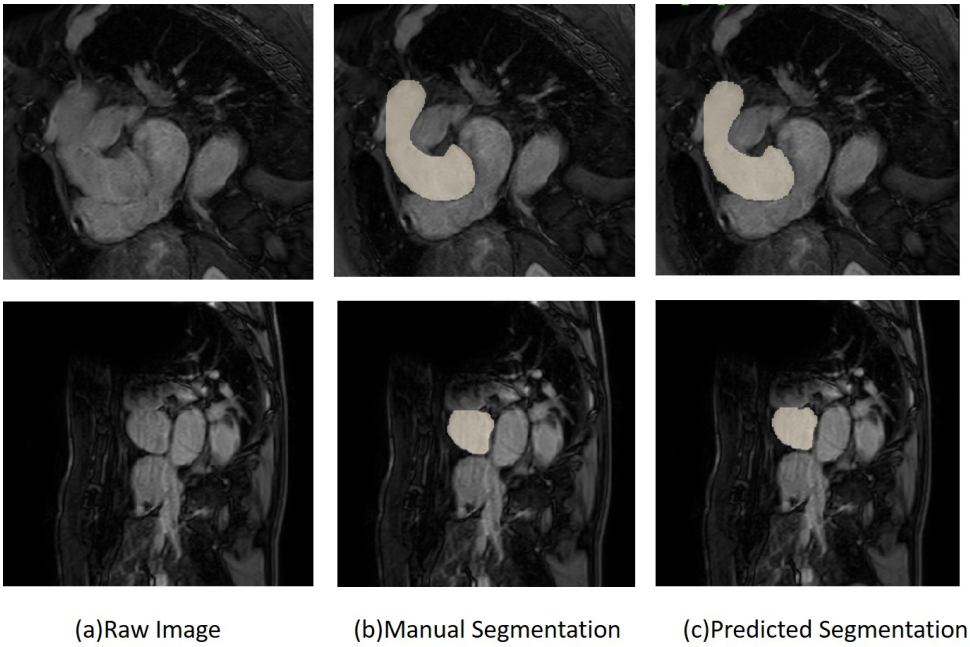


Figure 6.8: Two cases with low AI-based uncertainty (0.085 and 0.080) and high human-based uncertainty (level 1). The tissue is the Aor.

AI model. Thus, the AI-based uncertainty is considerably low. However, the human-based uncertainties for these slices are equal to 0 or 1, indicating that clinicians themselves were uncertain when performing annotations. Clinician

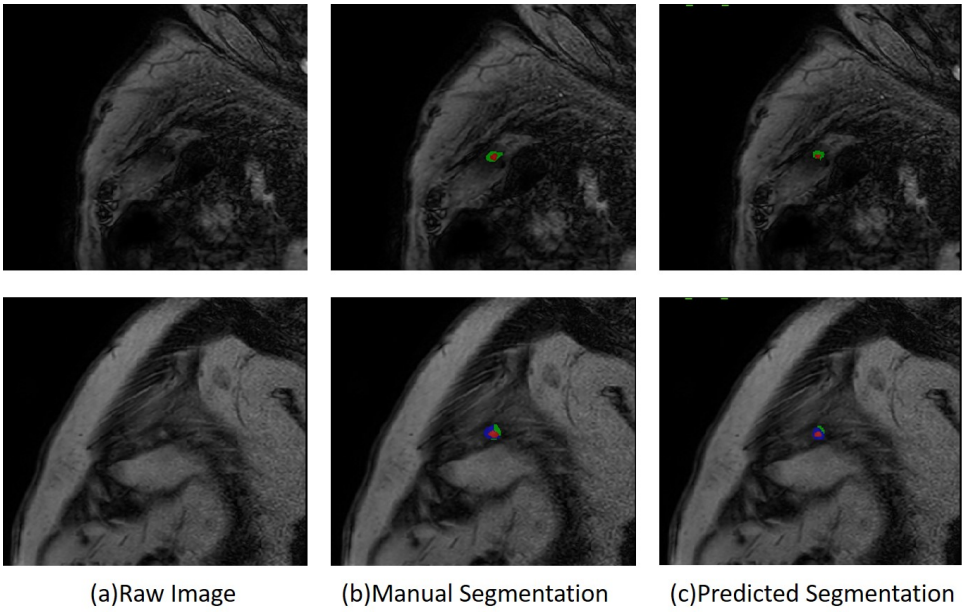


Figure 6.9: Two cases with low AI-based uncertainty (0.874 and 0.821) and high human-based uncertainty (level 2). The tissues are LV-En (red colour), LV-Ep (green colour), and Scar (Blue colour).

feedback suggested that the level of uncertainty pertained predominately to the presence of other labels such as the LV endocardium or myocardium in the given slice due to off-axis acquisition and image quality. Moreover, clinicians were more uncertain, especially in regions with close proximity of the aorta to surrounding structures.

Moreover, Figure 6.9 demonstrates the structures in each slice are difficult to distinguish as their sizes are too small. Hence the AI model struggled to achieve accurate segmentation, resulting in a significantly high level of uncertainty. However, the clinicians felt confident about their annotations. Clinicians suggested that their certainty was not derived from the present slice image but from the previous one. Their confidence was based on the accuracy of the prior slices and the fact that subsequent slices did not demonstrate anatomical displacement, thus inferring that the current slice is certain. Furthermore, considering that the current slice's location is at the bottom of the heart, the absence of any scar or other geometric complexities mitigates potential uncertain factors.

Hence, when clinicians made annotations, their uncertainty was determined based not only on the given slice but also contextual information. However, the proposed AI-based uncertainty algorithm is exclusively based on the information provided by the given slice image, without taking contextual factors into consideration. This is the key difference between human-based uncertainty and AI-based uncertainty investigated in this chapter.

6.6 Discussion

Feedback from clinicians suggests that pixel-level uncertainty as a visual map provides a quick visual understanding of an accurate spread of uncertainty. However, pixel-level uncertainty is the least reflective of segmentation quality as each image consists of numerous pixels. Class-level uncertainty provides valuable insights, such as identifying the classes that demonstrate good segmentation performance and filtering those that require further review. Moreover, according to the quantitative analysis, class-level uncertainty also outperforms other types of uncertainty. This is due to the fact all the different class-level uncertainties have an impact at image-level and slice-level uncertainties thus inaccurately skewing results if a single class is missing or inaccurately segmented.

On the other hand, the proposed quality quantification algorithm leverages uncertainty to infer the segmentation performance and improve the credibility of segmentation models for clinical practice. This supports the critical role of accurately defining uncertainty in quality quantification methods. Hence, class-level uncertainty is more suitable when evaluating the segmentation quality as it is calculated by the proposed fuzzy-uncertainty algorithm directly. Note that for some special scenarios, if clinicians are more interested in the segmentation quality of individual slices, slice-class-level uncertainty is the most appropriate

metric.

When it comes to human-based uncertainty and AI-based uncertainty, AI-based uncertainty lacks an understanding of the underlying geometry and is solely based on the provided data, whereas clinicians possess prior knowledge. This disparity highlights a potential difference between human uncertainty and AI uncertainty. Therefore, it is a promising direction to narrow this difference by adding human experience to the AI model or facilitating mutual learning between humans and AI. Note that the two clinicians who participated in the qualitative analysis are responsible for the annotation of ground truth images and human-based uncertainty. Thus they possess a more comprehensive and direct understanding of the provided cardiac MRI dataset.

6.7 Summary

In this chapter, to address objective 4, quantitative and qualitative analyses are conducted to investigate the application of uncertainty in clinical settings and the difference between human-based uncertainty and AI-based uncertainty. First, a real-world multi-slice 2D cardiac MRI dataset is collected from Nottingham University Hospital and Leeds Institute of Cardiovascular and Metabolic Medicine. Then the proposed AI-based uncertainty algorithm is applied to calculate the uncertainty of this dataset. Next, the relationship between true segmentation quality measured by DC and AI-based uncertainty is analyzed. Experimental results suggest that the proposed AI-based uncertainty has a strong correlation with the true segmentation quality of the cardiac MRI dataset. Furthermore, experiments were performed to explore human-based uncertainty as the clinicians provided their level of confidence during ground-truth annotation. According to the experimental results, high uncertainty labelled by humans

does not necessarily indicate low segmentation quality by AI-based segmentation models. For some specific structures (LV-Ep, RV-EP, and Pap), AI-based uncertainty has high agreements with human-based uncertainty.

Clinicians from the Nottingham University Hospitals NHS Trust took part in the qualitative analysis. Based on their feedback and comments, class-level uncertainty provides more detailed information and is more suitable to infer segmentation quality in comparison to other uncertainties. Pixel-level uncertainty as a visual map rapidly highlights the areas of uncertainty, but does not correlate with the segmentation performance directly. Representative cases with a great difference between human-based uncertainty and AI-based uncertainty are visualized to discuss with the clinicians. Feedback identify that humans utilise prior information (e.g. structural information) to formulate uncertainty scores. AI models lack the luxury of such prior learned knowledge highlighting this as a significant difference between the proposed AI-based uncertainty and human-based uncertainty. Potential future work should aim to incorporate human experience (e.g., the size of a cardiac model, structures' location, and contextual information) into AI models to improve the AI's understanding from a human perspective.

Having completed the quantitative and qualitative analyses on the application of AI-based uncertainty in clinical applications and the difference between human-based uncertainty and AI-based uncertainty, we will now proceed to the final conclusions of this thesis.

Chapter 7

Conclusions

This chapter summarizes the main points and contributions of this thesis. Limitations and future work are also discussed in this chapter.

7.1 Thesis Summary

Chapter 1 pointed out that the aim of this thesis was to improve the workflow of semantic segmentation through a combination of reducing model complexity, improving segmentation accuracy, and making semantic segmentation results more reliable and robust. To achieve this aim, the corresponding objectives were identified as follows:

[1] *Create a modeling framework in which the number of parameters is satisfyingly low*

It is widely recognized that to achieve satisfying segmentation performance, the semantic segmentation models normally have numerous learnable parameters. Moreover, the semantic segmentation models extract and aggregate spatial information and channel-wise features simultaneously.

These two facts increase the model's size and complexity. Hence, it is useful to design a novel module that can make it easy to achieve semantic segmentation.

[2] *Improve overall segmentation accuracy with particular emphasis on boundaries*

Many past and current methods investigate pixel-wise and region-wise losses while boundary-wise loss is underexplored. It is well known that one of the key aims of semantic segmentation is to precisely delineate objects' boundaries. Thus, it is critical to design a loss boundary-wise loss function that measures the errors around objects' boundaries and then improves the boundary accuracy of semantic segmentation.

[3] *Enhance interpretability of semantic segmentation results*

Without interpretability, the semantic segmentation results are inconvincible and not accepted by users, especially clinicians, which limits the application of semantic segmentation. Hence, in order to make semantic segmentation results more reliable and robust, a novel quality quantification algorithm is designed to enhance the interpretability of semantic segmentation results. The proposed quality quantification algorithm can help interpret the semantic segmentation results by segmentation uncertainty. Uncertainty has a negative relationship with segmentation quality, and fuzzy sets are an efficient and useful technique to handle and quantify uncertainty. It is a promising idea to leverage fuzzy sets to calculate the segmentation uncertainty and, therefore to indicate the quality of semantic segmentation results.

[4] *Evaluating the framework through real-world experimental studies*

After completing the algorithms or models design in the lab, the next step is to utilize the proposed algorithms and models to handle practical issues.

Hence, quantitative analysis and qualitative analysis are conducted with clinicians using a real-world cardiac MRI dataset to evaluate the framework.

7.2 Contributions and Publications

In Chapter 2, detailed background material and an overview of the literature this thesis uses and refers to are provided. This chapter is constituted of two parts: semantic segmentation and fuzzy methods. The definition of semantic segmentation, widely-used semantic segmentation models, semantic segmentation loss functions, and quality quantification algorithms for semantic segmentation are reviewed at the beginning of Chapter 2. Next, a brief introduction of various fuzzy methods adopted to deal with semantic segmentation issues is provided. First, the definition and representation methods of fuzzy sets are described in detail since they play a significant role in the proposed new fuzzy-based quality quantification algorithm. Then some basic theoretical knowledge and definitions of fuzzy rough sets are introduced, which constructs a solid mathematical support for the derivation procedure of the boundary-wise loss function. Finally, the definitions and formulas of different fuzzy integral operators are discussed in detail. Additionally, Chapter 2 also discusses the research gap in semantic segmentation and why these fuzzy-based technologies are useful.

- Contribution 1: A novel fuzzy integral layer in semantic segmentation models

  Chapter 3 addressed objective 1 by designing a new fuzzy integral layer. This layer can be integrated into semantic segmentation models and consists of a dimensionality reduction operator and OWA fusion operators. The dimensionality reduction operator is a convolutional layer, which can

help reduce the number of feature maps without affecting the process of loss back-propagation. OWA fusion operators are executed to generate new feature maps by fusing along the feature channel dimension. With the pre-defined parameters in OWA fusion operators and the application of a dimensionality reduction operator, the complexity of the segmentation model is dropped. Compared to the other fuzzy-integral-based semantic segmentation model, the proposed module is more efficient and achieves better segmentation performance. Moreover, it has been illustrated that when fuzzy integral modules are inserted in the encoding process of the UNet model, the UNet model complexity is considerably reduced while the segmentation performance remains similar as presented in our study [69] below.

[69] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "FuzzyDCNN: Incorporating Fuzzy Integral Layers to Deep Convolutional Neural Networks for Image Segmentation" in Proceedings IEEE International Conference on Fuzzy Systems, 2021, pp. 1-7.

Given a pre-defined semantic segmentation model, the related parameters are estimated by minimizing the given loss function iteratively in a training process. Thus, The loss function plays a considerably important role in the training process of semantic segmentation networks as it instigates the convergence process and affects the performance of neural networks.

- Contribution 2: A boundary-wise loss function for semantic segmentation based on fuzzy rough sets

  Chapter 4 addressed objective 2 by proposing a novel boundary-wise loss namely FRSLoss that can be used in various semantic segmentation models to improve the boundary accuracy of semantic segmentation. The FRSLoss is derived from the lower approximation of fuzzy rough sets.

The inclusion of the Gaussian kernel in the proposed FRSLoss formula allows us to normalize the boundary difference between the predicted segmentation and the actual segmentation, which stabilizes the convergence procedure and consumes less time. Considering the non-convex nature of the lower approximation of fuzzy rough sets, the distance transform algorithm is applied to calculate the FRSLoss in semantic segmentation tasks. Experimental studies showed that the proposed fuzzy rough sets loss outperforms other boundary-wise losses in terms of segmentation accuracy and time complexity. Compared with the commonly used pixel-wise and region-wise losses, the proposed boundary-wise loss has a similar performance but pays more attention to the boundaries as presented in our study [78] below.

[78] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "Boundary-wise Loss for Medical Image Segmentation Based on Fuzzy Rough Sets", Information Sciences (Under Review)

After completing the training process with the proposed novel FRSLoss function, the next step for semantic segmentation is to use the pre-trained segmentation model to segment new images. One challenge is that there are no ground truth images to help explain whether the segmentation results are reliable.

- Contribution 3: A novel quality quantification algorithm based on fuzzy sets

  Chapter 5 addressed objective 3 by proposing a novel fuzzy-uncertainty-based quality quantification algorithm. The quality quantification algorithm can quantify the quality of the predicted segmentation results as part of the model inference process and improve the understanding of the limitations of semantic segmentation through the explicit representation of uncertainty. Firstly, test-time augmentation and Monte Carlo dropout

are applied simultaneously to capture both the data and model uncertainties of the trained image segmentation model. Then a fuzzy set is generated to describe the captured uncertainty with the assistance of the linear Euclidean distance transform algorithm. Finally, the fuzziness of the generated fuzzy set is adopted to calculate an image-level segmentation uncertainty and therefore to infer the segmentation quality. The experimental results show that the estimated image-level uncertainties using the proposed method have strong correlations with the segmentation qualities measured by the Dice coefficient and outperform other five state-of-the-art quality quantification methods in classifying the segmentation results into good and poor quality groups–as presented in our studies [92, 91] below.

[92] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "Quality Quantification in Deep Convolutional Neural Networks for Skin Lesion Segmentation using Fuzzy Uncertainty Measurement, " in Proceedings IEEE International Conference on Fuzzy Systems, 2022, pp. 1-8.

[91] Q. Lin, X. Chen, C. Chen and J.M. Garibaldi, "A Novel Quality Control Algorithm for Medical Image Segmentation Based on Fuzzy Uncertainty". IEEE Transactions on Fuzzy Systems, 2022.

To further investigate the practical application of the proposed fuzzy uncertainty in a clinical setting, a real-world medical case study is conducted.

- Contribution 4: Quantitative analysis and qualitative analysis with clinicians using a real-world cardiac MRI dataset are implemented to investigate the practical application of uncertainty.

  Chapter 6 addressed objective 4 by conducting quantitative analysis and qualitative analysis using a real-world cardiac MRI dataset. Quantitative analysis is to obtain some experimental results based on the dataset, while

qualitative analysis is designed for clinicians to obtain feedback and comments regarding the application of uncertainty in real-world settings and how they annotate the uncertainty. This chapter primarily explores two questions: Firstly, how to use the proposed quality quantification algorithm to enhance the interpretability of semantic segmentation, thus enabling clinicians to better understand the cardiac MRI segmentation process. Secondly, the difference between human-based uncertainty and AI-based uncertainty is examined–as presented in our studies [116] below.

[116] Q. Lin, X. Chen, C. Chen, N. Jathanna, S. Jamil-Copley and J.M. Garibaldi, "Study of Uncertainty of AI and Human in Cardiac MRI Segmentation", Journal of Cardiovascular Magnetic Resonance (Under Review)

7.3 Limitations

The limitations of our work in this thesis are listed in this section.

The ordered weight averaging fusion method

In Chapter 3, a special case of fuzzy integrals namely ordered weight averaging (OWA) is utilized to merge channel information. According to Section 3.2, the OWA operator consists of two steps: 1) sort all feature maps in descending order; 2) aggregate the reordered feature maps with predefined OWA weights. The first step is time-consuming when there are numerous feature maps and the fusion weights are predefined which possibly limits the learning ability of the model. However, the sorting process is necessary since it ensures the nonlinear aggregation of OWA algorithms.

The test-time augmentation operators

In Chapter 5, we only used data augmentation during the test time as the proposed method is based on the assumption that the quality quantification algorithm is independent of the model training process and they do not share a similar data augmentation process. If the training and testing processes use similar data augmentation operators such as scaling, rotation, and flipping, it is expected to have a low uncertainty score of the output, which could impact the performance of the proposed quality quantification algorithm. Moreover, the selection of test-time augmentation operators is highly dependent on the dataset and application. For instance, rotation is not as effective as scaling and translation for the lung X-ray dataset, as all images are pretty much vertically aligned. The same conclusion would not be valid for other datasets. Therefore, one limitation is the use of test-time augmentation operators.

The distance transform method

In Chapter 4, the utilization of the distance transform algorithm in the FRSLoss function addresses the non-convex problem and makes the proposed loss applicable to the segmentation models. However, it is important to note that the distance transform algorithm has a limitation: the calculation process of the minimum distance is time-consuming, resulting in a longer convergence time for segmentation models employing the proposed FRSLoss in comparison to pixel-wise and region-wise losses.

Limited number of clinicians

In Chapter 6, given that the cardiac MRI dataset has been annotated by a pair of clinicians, the qualitative analysis is conducted exclusively by these two clinicians. The limited number of clinicians potentially leads to a confined interpretation rather than a universally applicable conclusion.

7.4 Future Work

Some useful and significant research directions that merit further research are discussed in this section.

Further investigate the boundary-wise loss

Although our suggested boundary-wise loss, FRSLoss, has achieved higher boundary accuracy than region-wise losses and pixel-wise losses, FRSLoss has similar or poorer performance in contrast to other losses for some segmentation metrics. Furthermore, our current work only takes signal loss into consideration. In theory, the combination of pixel-wise loss, region-wise loss and boundary-wise loss can deal with the pixel, region and boundary simultaneously. Therefore, whether using a combined loss to train the semantic segmentation model is beneficial for the improvement of segmentation performance on all segmentation metrics is still an open question and worth further investigation.

Exploration of uncertainty estimation methods

In the proposed fuzzy-uncertainty-based quality quantification algorithm, Test-time augmentation and Monte Caro Dropout are adopted to capture data un-

certainty and model uncertainty, respectively. It should be noted that the data augmentation operators of the training process may have an impact on the uncertainty calculation in the proposed quality quantification method. The widely-used data augmentation operators consist of scaling, rotation, flipping, etc. The objective of the experiments in Chapter 5 is to compare the proposed fuzzy-uncertainty-based quality quantification algorithm with other uncertainty-based methods. The TTA and MCdropout operators are the same for all uncertainty-based methods. However, it is not clear which data augmentation transformation has the most significant effect on data uncertainty. Moreover, except for Monte Caro Dropout, there are other methods to capture the model uncertainty, e.g. ensemble [26] and MIMO [127]. Therefore, in the future, it would be interesting to explore the different data augmentation operators and model uncertainty capturing algorithms.

On the other hand, our latest work [126] investigates the efficiency of interval fuzzy sets in uncertainty estimation. Experimental results show that interval-fuzzy-sets-based uncertainty is useful for detecting out-of-distribution data. However, this work is an initial step for the application of interval fuzzy sets. Only a single dataset and basic interval fuzzy sets generation method are applied in the corresponding experiments. More sophisticated techniques, datasets, and applications can be explored in future work.

Comparisons of the performance of the proposed quality quantification algorithms on different semantic segmentation models

In Chapter 5, we only investigate the performance of the fuzzy-uncertainty-based quality quantification algorithm on the 2D UNet model and 3D VNet

model. In fact, there are numerous deep convolutional neural networks (DCNN)-based segmentation models including FCN [13], SegNet [2], DeepLab [14] etc. It is useful to apply our suggested approach to other commonly-used semantic segmentation models in the future to broaden its application scope.

Moreover, with the advent of the transformer module [128] in 2017, many researchers adopted the transformer module to replace some convolutional layers and achieved outstanding segmentation performance, such as ViT [129] and Swin transformer [130]. Currently, there is no research regarding the uncertainty of transformer-based segmentation models. It is a promising and worthy topic to investigate the performance of the proposed quality quantification algorithms on transformer models

Incorporate experts' experience into AI models

As AI models normally have limited knowledge and unawareness of the intricacies of that specific location, there is a potential difference between the proposed AI-based uncertainty and human-based uncertainty. In the future, the experts' experience e.g. the size of a heart model, tissues' location, and contextual information is possible to incorporate into the AI model, which potentially narrows the difference between human-based uncertainty and AI-based uncertainty and improves the performance of AI models.

Involve more clinicians when conducting qualitative analysis

To further investigate human-based uncertainty, more clinicians who are responsible for annotation should be involved. It contributes to obtaining general and convincing conclusions, thereby improving the performance of semantic segmentation.

Bibliography

[1] 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*, pp. 234–241, Springer, 2015.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[3] J. Mendel, H. Hagras, W.-W. Tan, W. W. Melek, and H. Ying, *Introduction to type-2 fuzzy logic control: theory and applications*. John Wiley & Sons, 2014.

[4] J. M. Mendel, "Uncertain rule-based fuzzy systems," *Introduction and new directions*, p. 684, 2017.

[5] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE transactions on fuzzy systems*, vol. 14, no. 6, pp. 808–821, 2006.

[6] S. R. Price, S. R. Price, and D. T. Anderson, "Introducing fuzzy layers for deep learning," in *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–6, IEEE, 2019.

[7] K. J. Batenburg and J. Sijbers, "Adaptive thresholding of tomograms by projection distance minimization," *Pattern Recognition*, vol. 42, no. 10, pp. 2297–2305, 2009.

[8] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 26, no. 11, pp. 1452–1458, 2004.

[9] T. Lindeberg and M.-X. Li, "Segmentation and classification of edges using minimum description length approximation and complementary junction cues," *Computer Vision and Image Understanding*, vol. 67, no. 1, pp. 88–98, 1997.

[10] J. Fu, S. Lee, S. Wong, J. Yeh, A. Wang, and H. Wu, "Image segmentation feature selection and pattern classification for mammographic microcalcifications," *Computerized Medical Imaging and Graphics*, vol. 29, no. 6, pp. 419–429, 2005.

[11] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[12] J. Jiang, Y.-c. Hu, C.-J. Liu, D. Halpenny, M. D. Hellmann, J. O. Deasy, G. Mageras, and H. Veeraraghavan, "Multiple resolution residually connected feature streams for automatic lung tumor segmentation from ct images," *IEEE transactions on medical imaging*, vol. 38, no. 1, pp. 134–144, 2018.

[13] 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*, pp. 3431–3440, 2015.

[14] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille,

"Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[15] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[16] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[17] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

[18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

[19] M. Yi-de, L. Qing, and Q. Zhi-Bai, "Automated image segmentation using improved pcnn model based on cross-entropy," in *Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004.*, pp. 743–746, IEEE, 2004.

[20] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," in *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pp. 240–248, Springer, 2017.

[21] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of*

the IEEE international conference on computer vision, pp. 1395–1403, 2015.

[22] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[23] S. A. Taghanaki, Y. Zheng, S. K. Zhou, B. Georgescu, P. Sharma, D. Xu, D. Comaniciu, and G. Hamarneh, "Combo loss: Handling input and output imbalance in multi-organ segmentation," *Computerized Medical Imaging and Graphics*, vol. 75, pp. 24–33, 2019.

[24] D. Dubois and H. Prade, "Rough fuzzy sets and fuzzy rough sets," *International Journal of General System*, vol. 17, no. 2-3, pp. 191–209, 1990.

[25] S. Hao, Y. Zhou, and Y. Guo, "A brief survey on semantic segmentation with deep learning," *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[26] A. Mehrtash, W. M. Wells, C. M. Tempany, P. Abolmaesumi, and T. Kapur, "Confidence calibration and predictive uncertainty estimation for deep medical image segmentation," *IEEE transactions on medical imaging*, vol. 39, no. 12, pp. 3868–3878, 2020.

[27] L. A. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965.

[28] S. K. De, R. Biswas, and A. R. Roy, "An application of intuitionistic fuzzy sets in medical diagnosis," *Fuzzy sets and Systems*, vol. 117, no. 2, pp. 209–213, 2001.

[29] K.-C. Kwak and W. Pedrycz, "Face recognition using a fuzzy fisherface classifier," *Pattern recognition*, vol. 38, no. 10, pp. 1717–1732, 2005.

[30] X. Wang, H. Zhang, and Z. Xu, "Public sentiments analysis based on fuzzy logic for text," *International Journal of Software Engineering and Knowledge Engineering*, vol. 26, no. 09n10, pp. 1341–1360, 2016.

[31] X. Chen, R. Girshick, K. He, and P. Dollár, "Tensormask: A foundation for dense object segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2061–2069, 2019.

[32] F. Hafiz, A. Shafie, O. Khalifa, and M. Ali, "Foreground segmentation-based human detection with shadow removal," in *International Conference on Computer and Communication Engineering (ICCCE'10)*, pp. 1–6, IEEE, 2010.

[33] Y. Lin, J. Shen, Y. Wang, and M. Pantic, "Roi tanh-polar transformer network for face parsing in the wild," *Image and Vision Computing*, vol. 112, p. 104190, 2021.

[34] G. Wang, X. Qin, J. Shen, Z. Zhang, D. Han, and C. Jiang, "Quantitative analysis of microscopic structure and gas seepage characteristics of low-rank coal based on ct three-dimensional reconstruction of ct images and fractal theory," *Fuel*, vol. 256, p. 115900, 2019.

[35] T. Lei, P. Liu, X. Jia, X. Zhang, H. Meng, and A. K. Nandi, "Automatic fuzzy clustering framework for image segmentation," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 9, pp. 2078–2092, 2019.

[36] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, pp. 1520–1528, 2015.

[37] C. Wang, T. MacGillivray, G. Macnaught, G. Yang, and D. Newby, "A two-stage 3d unet framework for multi-class segmentation on full resolution image," *arXiv preprint arXiv:1804.04341*, 2018.

[38] S. Jadon, "A survey of loss functions for semantic segmentation," in *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–7, IEEE, 2020.

[39] V. Pihur, S. Datta, and S. Datta, "Weighted rank aggregation of cluster validation measures: a monte carlo cross-entropy approach," *Bioinformatics*, vol. 23, no. 13, pp. 1607–1615, 2007.

[40] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, S. P. Prabhu, S. K. Warfield, and A. Gholipour, "Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: Application to multiple sclerosis lesion detection," *IEEE Access*, vol. 7, pp. 1721–1735, 2018.

[41] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, "Tversky loss function for image segmentation using 3d fully convolutional deep networks," in *International workshop on machine learning in medical imaging*, pp. 379–387, Springer, 2017.

[42] N. Abraham and N. M. Khan, "A novel focal tversky loss function with improved attention u-net for lesion segmentation," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pp. 683–687, IEEE, 2019.

[43] D. Karimi and S. E. Salcudean, "Reducing the hausdorff distance in medical image segmentation with convolutional neural networks," *IEEE Transactions on medical imaging*, vol. 39, no. 2, pp. 499–513, 2019.

[44] T. Kohlberger, V. Singh, C. Alvino, C. Bahlmann, and L. Grady, "Evaluating segmentation error without ground truth," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 528–536, Springer, 2012.

[45] R. Robinson, O. Oktay, W. Bai, V. V. Valindria, M. M. Sanghvi, N. Aung, J. M. Paiva, F. Zemrak, K. Fung, E. Lukaschuk, *et al.*, "Real-time prediction of segmentation quality," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 578–585, Springer, 2018.

[46] S. Wang, G. Tarroni, C. Qin, Y. Mo, C. Dai, C. Chen, B. Glocker, Y. Guo, D. Rueckert, and W. Bai, "Deep generative model-based quality control for cardiac mri segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 88–97, Springer, 2020.

[47] K. Hoebel, V. Andrearczyk, A. Beers, J. Patel, K. Chang, A. Depeursinge, H. Müller, and J. Kalpathy-Cramer, "An exploration of uncertainty information for segmentation quality assessment," in *Medical Imaging 2020: Image Processing*, vol. 11313, p. 113131K, International Society for Optics and Photonics, 2020.

[48] A. G. Roy, S. Conjeti, N. Navab, C. Wachinger, A. D. N. Initiative, *et al.*, "Bayesian quicknat: Model uncertainty in deep whole-brain segmentation for structure-wise quality control," *NeuroImage*, vol. 195, pp. 11–22, 2019.

[49] A. G. Roy, S. Conjeti, N. Navab, and C. Wachinger, "Inherent brain segmentation quality control from fully convnet monte carlo sampling," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 664–672, Springer, 2018.

[50] T. DeVries and G. W. Taylor, "Leveraging uncertainty estimates for predicting segmentation quality," *arXiv preprint arXiv:1807.00502*, 2018.

[51] M. R. Rajati and J. M. Mendel, "On advanced computing with words

using the generalized extension principle for type-1 fuzzy sets," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 5, pp. 1245–1261, 2013.

[52] A. V. Chernov, M. A. Butakova, V. A. Bogachev, V. D. Vereskun, and A. N. Guda, "Study of fuzzy sets similarity and its application in intelligent transportation systems," *Global Journal of Pure and Applied Mathematics*, vol. 12, no. 6, p. 5095, 2016.

[53] G. Klir and T. Folger, "Fuzzy sets: Uncertainty and information, prentice," 1988.

[54] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—i," *Information sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[55] Q. Jun, H. Dinçer, and S. Yüksel, "Stochastic hybrid decision-making based on interval type 2 fuzzy sets for measuring the innovation capacities of financial institutions," *International Journal of Finance & Economics*, vol. 26, no. 1, pp. 573–593, 2021.

[56] M. Deveci, U. Cali, S. Kucuksari, and N. Erdogan, "Interval type-2 fuzzy sets based multi-criteria decision-making model for offshore wind farm development in ireland," *Energy*, vol. 198, p. 117317, 2020.

[57] Q. Jiang, X. Jin, J. Hou, S.-J. Lee, and S. Yao, "Multi-sensor image fusion based on interval type-2 fuzzy sets and regional features in nonsubsampled shearlet transform domain," *IEEE Sensors Journal*, vol. 18, no. 6, pp. 2494–2505, 2018.

[58] Y. Dorfeshan, S. M. Mousavi, E. K. Zavadskas, and J. Antucheviciene, "A new enhanced aras method for critical path selection of engineering projects with interval type-2 fuzzy sets," *International Journal of Information Technology & Decision Making*, vol. 20, no. 01, pp. 37–65, 2021.

[59] Z. Pawlak, "Rough sets," *International journal of computer & information sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[60] Q. Hu, L. Zhang, D. Chen, W. Pedrycz, and D. Yu, "Gaussian kernel based fuzzy rough sets: model, uncertainty measures and applications," *International Journal of Approximate Reasoning*, vol. 51, no. 4, pp. 453–471, 2010.

[61] J.-S. Mi and W.-X. Zhang, "An axiomatic characterization of a fuzzy generalization of rough sets," *Information Sciences*, vol. 160, no. 1-4, pp. 235–249, 2004.

[62] D. S. Yeung, D. Chen, E. C. Tsang, J. W. Lee, and W. Xizhao, "On the generalization of fuzzy rough sets," *IEEE Transactions on fuzzy systems*, vol. 13, no. 3, pp. 343–361, 2005.

[63] G. Choquet, "Theory of capacities," in *Annales de l'institut Fourier*, vol. 5, pp. 131–295, 1954.

[64] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Transactions on systems, Man, and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

[65] R. R. Yager, "Families of owa operators," *Fuzzy sets and systems*, vol. 59, no. 2, pp. 125–148, 1993.

[66] M. O'Hagan, "Fuzzy decision aids," in *21th Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 624–628, IEEE and Maple Press, 1987.

[67] R. R. Yager and D. P. Filev, "Parameterized and-uke and or-like owa operators," *International Journal of General System*, vol. 22, no. 3, pp. 297–316, 1994.

[68] Z. Xu, "An overview of methods for determining owa weights," *International journal of intelligent systems*, vol. 20, no. 8, pp. 843–865, 2005.

[69] Q. Lin, X. Chen, C. Chen, and J. M. Garibaldi, "Fuzzydcnn: Incorporating fuzzy integral layers to deep convolutional neural networks for image segmentation," in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–7, IEEE, 2021.

[70] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[71] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European conference on computer vision*, pp. 483–499, Springer, 2016.

[72] M. Sugeno, "Fuzzy measure and fuzzy integral," *Transactions of the Society of Instrument and Control Engineers*, vol. 8, no. 2, pp. 218–226, 1972.

[73] R. Yang and R. Ouyang, "Classification based on choquet integral," *Journal of Intelligent & Fuzzy Systems*, vol. 27, no. 4, pp. 1693–1702, 2014.

[74] B. Hadjadji and Y. Chibani, "Two combination stages of clustered one-class classifiers for writer identification from text fragments," *Pattern Recognition*, vol. 82, pp. 147–162, 2018.

[75] D. T. Anderson, G. J. Scott, M. A. Islam, B. Murray, and R. Marcum, "Fuzzy choquet integration of deep convolutional neural networks for remote sensing," in *Computational Intelligence for Pattern Recognition*, pp. 1–28, Springer, 2018.

[76] G. J. Scott, R. A. Marcum, C. H. Davis, and T. W. Nivin, "Fusion of deep convolutional neural networks for land cover classification of high-resolution imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 9, pp. 1638–1642, 2017.

[77] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–156, 2006.

[78] Q. Lin, X. Chen, C. Chen, and J. M. Garibaldi, "Boundary-wise loss for medical image segmentation based on fuzzy rough sets," *Information Sciences*, 2023.

[79] V. Murali, "Fuzzy equivalence relations," *Fuzzy sets and systems*, vol. 30, no. 2, pp. 155–163, 1989.

[80] P. Maji and S. Paul, "Rough-fuzzy clustering for grouping functionally similar genes from microarray data," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 10, no. 2, pp. 286–299, 2012.

[81] Q. Chen, M. Huang, H. Wang, and G. Xu, "A feature discretization method based on fuzzy rough sets for high-resolution remote sensing big data under linear spectral model," *IEEE Transactions on Fuzzy Systems*, 2021.

[82] N. Mac Parthaláin, R. Jensen, and R. Diao, "Fuzzy-rough set bireducts for data reduction," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 8, pp. 1840–1850, 2019.

[83] Q. Hu, L. Zhang, Y. Zhou, and W. Pedrycz, "Large-scale multimodality attribute reduction with multi-kernel fuzzy rough sets," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 1, pp. 226–238, 2017.

[84] N. Mac Parthaláin and R. Jensen, "Fuzzy-rough feature selection using flock of starlings optimisation," in *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–8, IEEE, 2015.

[85] J. Goutsias and S. Batman, "Morphological methods for biomedical image analysis," *Handbook of medical imaging*, vol. 2, pp. 175–272, 2000.

[86] G. J. Grevera and J. K. Udupa, "Shape-based interpolation of multidimensional grey-level images," *IEEE transactions on medical imaging*, vol. 15, no. 6, pp. 881–892, 1996.

[87] G. Penney, J. Little, J. Weese, D. Hill, and D. Hawkes, "Deforming a preoperative volume to represent the intraoperative scene," *Computer Aided Surgery*, vol. 7, no. 2, pp. 63–73, 2002.

[88] M. Herk, "Image registration using chamfer matching," *Handbook of medical imaging processing and analysis*, 2000.

[89] P. F. Felzenszwalb and D. P. Huttenlocher, "Distance transforms of sampled functions," *Theory of computing*, vol. 8, no. 1, pp. 415–428, 2012.

[90] H. Irshad, L. Montaser-Kouhsari, G. Waltz, O. Bucur, J. Nowak, F. Dong, N. W. Knoblauch, and A. H. Beck, "Crowdsourcing image annotation for nucleus detection and segmentation in computational pathology: evaluating experts, automated methods, and the crowd," in *Pacific symposium on biocomputing Co-chairs*, pp. 294–305, World Scientific, 2014.

[91] Q. Lin, X. Chen, C. Chen, and J. M. Garibaldi, "A novel quality control algorithm for medical image segmentation based on fuzzy uncertainty," *IEEE Transactions on Fuzzy Systems*, 2022.

[92] Q. Lin, X. Chen, C. Chen, and J. M. Garibaldi, "Quality quantification in deep convolutional neural networks for skin lesion segmentation using

fuzzy uncertainty measurement," in *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–8, IEEE, 2022.

[93] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, 2021.

[94] T. Nair, D. Precup, D. L. Arnold, and T. Arbel, "Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation," *Medical image analysis*, vol. 59, p. 101557, 2020.

[95] N. R. Pal and J. C. Bezdek, "Measuring fuzzy uncertainty," *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 2, pp. 107–118, 1994.

[96] D. Wu and J. M. Mendel, "Uncertainty measures for interval type-2 fuzzy sets," *Information sciences*, vol. 177, no. 23, pp. 5378–5393, 2007.

[97] 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)*, pp. 565–571, IEEE, 2016.

[98] J. Schlüter and T. Grill, "Exploring data augmentation for improved singing voice detection with neural networks.," in *ISMIR*, pp. 121–126, 2015.

[99] G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, and T. Vercauteren, "Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks," *Neurocomputing*, vol. 338, pp. 34–45, 2019.

[100] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Rep-

resenting model uncertainty in deep learning," in *international conference on machine learning*, pp. 1050–1059, PMLR, 2016.

[101] D. E. Diamantis and D. K. Iakovidis, "Fuzzy pooling," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 11, pp. 3481–3488, 2020.

[102] E. Van Broekhoven and B. De Baets, "Fast and accurate center of gravity defuzzification of fuzzy system outputs defined on trapezoidal fuzzy partitions," *Fuzzy sets and systems*, vol. 157, no. 7, pp. 904–918, 2006.

[103] T. A. Runkler, C. Chen, and R. John, "Type reduction operators for interval type–2 defuzzification," *Information Sciences*, vol. 467, pp. 464–476, 2018.

[104] S. Jaeger, A. Karargyris, S. Candemir, L. Folio, J. Siegelman, F. Callaghan, Z. Xue, K. Palaniappan, R. K. Singh, S. Antani, *et al.*, "Automatic tuberculosis screening using chest radiographs," *IEEE transactions on medical imaging*, vol. 33, no. 2, pp. 233–245, 2013.

[105] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy, "Dataset of breast ultrasound images," *Data in brief*, vol. 28, p. 104863, 2020.

[106] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation.," *Nature methods*, vol. 9, no. 7, pp. 637–637, 2012.

[107] A. Hoopes, M. Hoffmann, B. Fischl, J. Guttag, and A. V. Dalca, "Hypermorph: Amortized hyperparameter learning for image registration," in *International Conference on Information Processing in Medical Imaging*, pp. 3–17, Springer, 2021.

[108] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[109] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[110] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise reduction in speech processing*, pp. 1–4, Springer, 2009.

[111] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[112] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.

[113] O. J. Dunn, "Multiple comparisons among means," *Journal of the American statistical association*, vol. 56, no. 293, pp. 52–64, 1961.

[114] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[115] J. M. Mendel, "Type-2 fuzzy sets and systems: an overview," *IEEE computational intelligence magazine*, vol. 2, no. 1, pp. 20–29, 2007.

[116] Q. Lin, X. Chen, C. Chen, J. Nikesh, J.-C. Shahnaz, and J. M. Garibaldi, "Study of uncertainty of ai and human in cardiac mri segmentation," *Medical Image Analysis*, 2023.

[117] E. Tülümen, B. Rudic, H. Ringlage, A. Hohneck, S. Röger, V. Liebe, J. Kuschyk, D. Overhoff, J. Budjan, I. Akin, *et al.*, "Extent of periinfarct scar on late gadolinium enhancement cardiac magnetic resonance imaging and outcome in patients with ischemic cardiomyopathy," *Heart Rhythm*, vol. 18, no. 6, pp. 954–961, 2021.

[118] K. Seetharam and S. Lerakis, "Cardiac magnetic resonance imaging: the future is bright," *F1000Research*, vol. 8, 2019.

[119] A. Demirkiran, H. Everaars, R. P. Amier, C. Beijnink, M. J. Bom, M. J. Götte, R. B. van Loon, J. L. Selder, A. C. van Rossum, and R. Nijveldt, "Cardiovascular magnetic resonance techniques for tissue characterization after acute myocardial injury," *European Heart Journal-Cardiovascular Imaging*, vol. 20, no. 7, pp. 723–734, 2019.

[120] R. D. Kociol, L. T. Cooper, J. C. Fang, J. J. Moslehi, P. S. Pang, M. A. Sabe, R. V. Shah, D. B. Sims, G. Thiene, O. Vardeny, *et al.*, "Recognition and initial management of fulminant myocarditis: a scientific statement from the american heart association," *Circulation*, vol. 141, no. 6, pp. e69–e92, 2020.

[121] N. Painchaud, Y. Skandarani, T. Judge, O. Bernard, A. Lalande, and P.-M. Jodoin, "Cardiac segmentation with strong anatomical guarantees," *IEEE transactions on medical imaging*, vol. 39, no. 11, pp. 3703–3713, 2020.

[122] M. Khened, V. A. Kollerathu, and G. Krishnamurthi, "Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers," *Medical image analysis*, vol. 51, pp. 21–45, 2019.

[123] Q. Yue, X. Luo, Q. Ye, L. Xu, and X. Zhuang, "Cardiac segmentation from lge mri using deep neural network incorporating shape and spatial priors," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*, pp. 559–567, Springer, 2019.

[124] L. Kuipers and H. Niederreiter, *Uniform distribution of sequences*. Courier Corporation, 2012.

[125] M. Mahsereci, L. Balles, C. Lassner, and P. Hennig, "Early stopping without a validation set," *arXiv preprint arXiv:1703.09580*, 2017.

[126] Q. Lin, X. Chen, C. Chen, P. Direnc, and J. M. Garibaldi, "Fuzzy uncertainty-based out-of-distribution detection algorithm for semantic segmentation," in *2023 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–6.

[127] D. Karimi and A. Gholipour, "Improving calibration and out-of-distribution detection in deep models for medical image segmentation," *IEEE Transactions on Artificial Intelligence*, 2022.

[128] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[129] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[130] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Appendices

The appendices present an analysis of three cases with varying segmentation quality (good, fair, and poor) to explore the relationship between AI-based uncertainty and segmentation quality. To evaluate the segmentation quality, the proposed fuzzy-uncertainty algorithm is applied to calculate the uncertainty for each case. We also measure the Dice for each case, which is calculated by comparing the predicted segmentation image to the ground truth image. Additionally, pre-trained regression models ($y = -0.826 \times x + 1.037$ for slice-class level and $y = -1.238 \times x + 0.975$ for image-class level) are utilized to calculate the predicted Dice for each case based on the uncertainty. Finally, the error is the discrepancy between the real dice and the predicted dice. The results of this analysis are presented in the following tables.

Appendix A

Good Segmentation Case

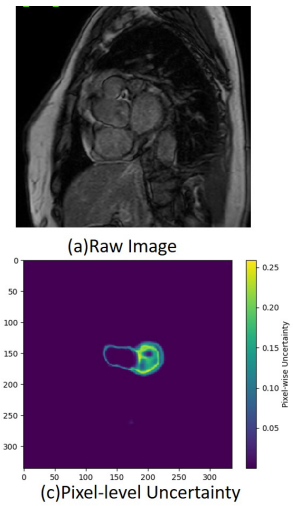
The segmentation case $20CA015\_N133\_SAX.nii.gz$ has the following shape: $11 \times 336 \times 336$ and its segmentation quality (measured by Dice) is 0.793.

A.1 Whole 3D Image

Table A1: The results for the whole 20CA015_N133_SAX.nii.gz Image

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.091	0.923	0.862	0.061
LV-Ep	0.142	0.845	0.798	0.047
Scar	0.360	0.699	0.529	0.170
RV-En	0.059	0.945	0.901	0.044
RV-Ep	0.257	0.716	0.657	0.059
Pap	0.347	0.611	0.545	0.066
Aor	0.130	0.813	0.814	0.001
Whole	0.198	0.793	0.729	0.064

A.2 Each Slice

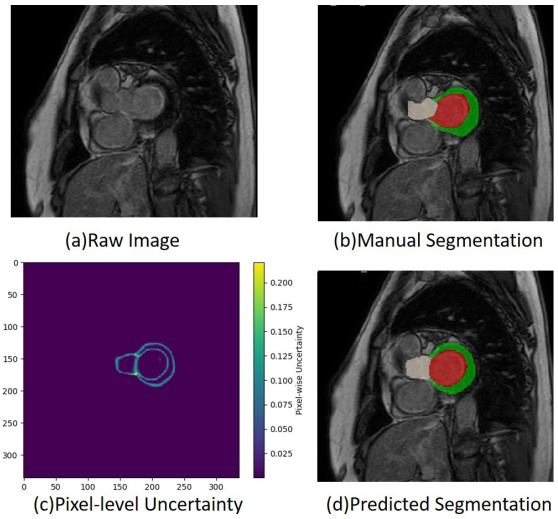


Slice 1

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.669	0	0.296	0.296
LV-Ep	0.659	0	0.307	0.307
Scar	NA	NA	NA	NA
RV-En	NA	NA	NA	NA
RV-Ep	NA	NA	NA	NA
Pap	NA	NA	NA	NA
Aor	0.162	0.834	0.832	0.002
Slice	0.213	0.690	0.778	0.088

(e) Uncertainty---Dice

Figure A1: The results for slice-1 of 20CA015_N133_SAX.nii.gz

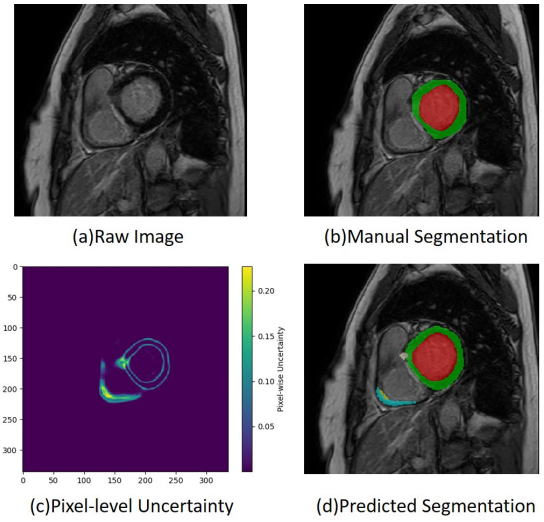


Slice 2

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.098	0.934	0.899	0.035
LV-Ep	0.210	0.792	0.781	0.011
Scar	NA	NA	NA	NA
RV-En	NA	NA	NA	NA
RV-Ep	NA	NA	NA	NA
Pap	NA	NA	NA	NA
Aor	0.146	0.828	0.849	0.021
Slice	0.065	0.936	0.934	0.002

(e) Uncertainty---Dice

Figure A2: The results for slice-2 of 20CA015_N133_SAX.nii.gz

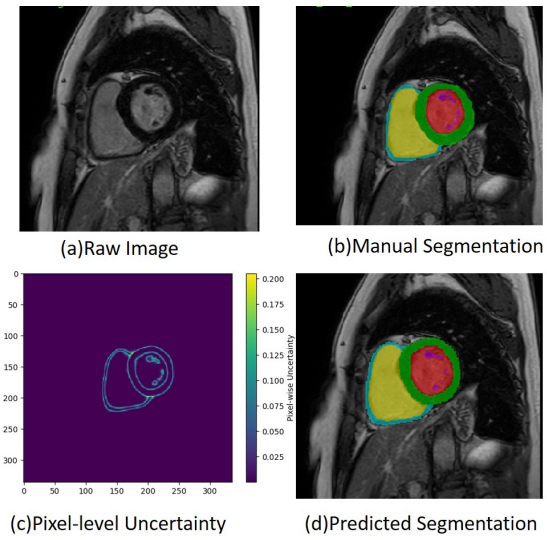


Slice 3

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.058	0.963	0.942	0.022
LV-Ep	0.194	0.893	0.799	0.095
Scar	NA	NA	NA	NA
RV-En	0.872	0.00	0.082	0.082
RV-Ep	0.576	0.00	0.394	0.394
Pap	NA	NA	NA	NA
Aor	0.691	0.00	0.273	0.273
Slice	0.341	0.551	0.642	0.091

(e) Uncertainty---Dice

Figure A3: The results for slice-3 of 20CA015_N133_SAX.nii.gz

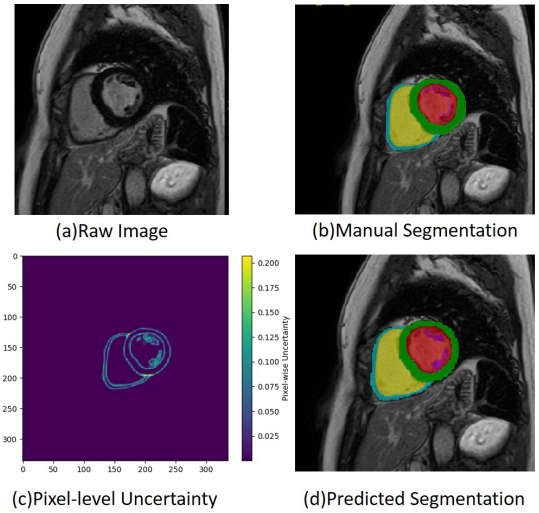


Slice 4

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.089	0.937	0.909	0.028
LV-Ep	0.123	0.872	0.873	0.0
Scar	NA	NA	NA	NA
RV-En	0.061	0.962	0.939	0.024
RV-Ep	0.186	0.769	0.806	0.037
Pap	0.344	0.656	0.639	0.018
Aor	NA	NA	NA	NA
Slice	0.115	0.885	0.882	0.003

(e) Uncertainty---Dice

Figure A4: The results for slice-4 of 20CA015_N133_SAX.nii.gz

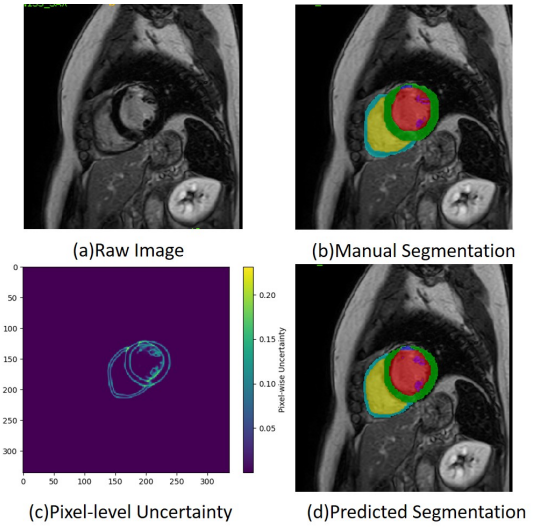


Slice 5

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.107	0.907	0.89	0.017
LV-Ep	0.145	0.871	0.85	0.022
Scar	NA	NA	NA	NA
RV-En	0.090	0.955	0.908	0.047
RV-Ep	0.233	0.766	0.757	0.009
Pap	0.364	0.538	0.618	0.08
Aor	NA	NA	NA	NA
Slice	0.862	0.862	0.861	0.001

(e) Uncertainty---Dice

Figure A5: The results for slice-5 of 20CA015_N133_SAX.nii.gz

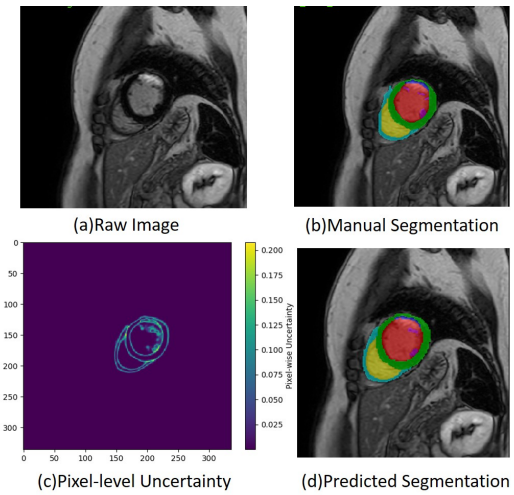


Slice 6

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.135	0.906	0.861	0.046
LV-Ep	0.191	0.840	0.801	0.039
Scar	0.381	0.533	0.601	0.067
RV-En	0.069	0.932	0.93	0.002
RV-Ep	0.276	0.740	0.712	0.035
Pap	0.365	0.704	0.617	0.088
Aor	NA	NA	NA	NA
Slice	0.202	0.809	0.789	0.020

(e) Uncertainty---Dice

Figure A6: The results for slice-6 of 20CA015_N133_SAX.nii.gz

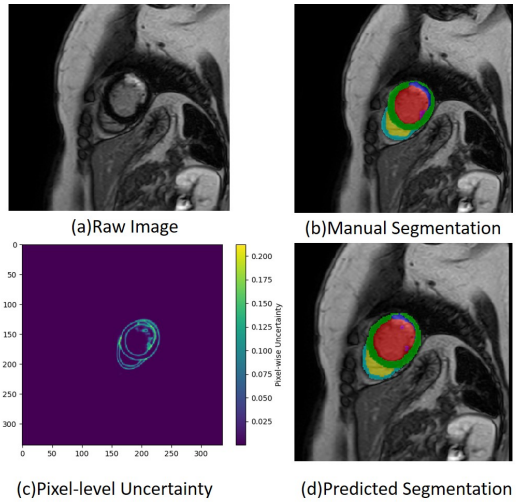


Slice 7

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.110	0.933	0.887	0.046
LV-Ep	0.165	0.854	0.829	0.026
Scar	0.430	0.856	0.549	0.307
RV-En	0.095	0.935	0.903	0.032
RV-Ep	0.228	0.747	0.762	0.015
Pap	0.438	0.564	0.54	0.025
Aor	NA	NA	NA	NA
Slice	0.209	0.841	0.782	0.059

(e) Uncertainty---Dice

Figure A7: The results for slice-7 of 20CA015_N133_SAX.nii.gz

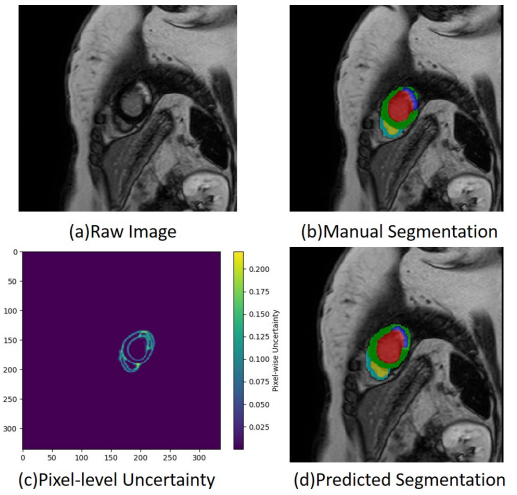


Slice 8

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.116	0.936	0.88	0.056
LV-Ep	0.199	0.869	0.793	0.077
Scar	0.466	0.596	0.511	0.086
RV-En	0.150	0.911	0.845	0.067
RV-Ep	0.234	0.755	0.756	0.0
Pap	0.481	0.438	0.494	0.056
Aor	NA	NA	NA	NA
Slice	0.235	0.786	0.755	0.031

(e) Uncertainty---Dice

Figure A8: The results for slice-8 of 20CA015_N133_SAX.nii.gz

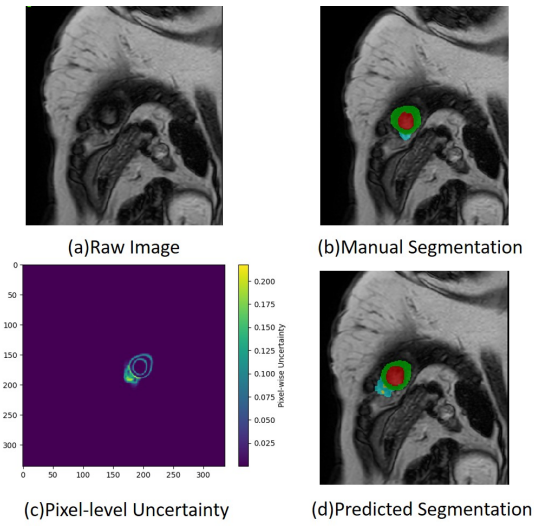


Slice 9

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.124	0.944	0.872	0.072
LV-Ep	0.234	0.803	0.756	0.047
Scar	0.413	0.771	0.567	0.205
RV-En	0.200	0.847	0.792	0.055
RV-Ep	0.365	0.670	0.617	0.053
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.190	0.862	0.802	0.060

(e) Uncertainty---Dice

Figure A9: The results for slice-9 of 20CA015_N133_SAX.nii.gz

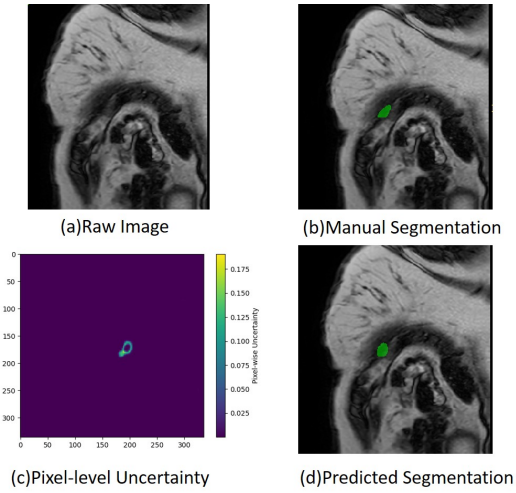


Slice 10

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.129	0.902	0.867	0.035
LV-Ep	0.204	0.849	0.788	0.061
Scar	NA	NA	NA	NA
RV-En	0.786	0.000	0.172	0.172
RV-Ep	0.608	0.107	0.36	0.253
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.246	0.694	0.742	0.048

(e) Uncertainty---Dice

Figure A10: The results for slice-10 of 20CA015_N133_SAX.nii.gz



Slice 11

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	NA	NA	NA	NA
LV-Ep	0.253	0.796	0.736	0.06
Scar	NA	NA	NA	NA
RV-En	NA	NA	NA	NA
RV-Ep	0.800	0.00	0.158	0.158
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.150	0.828	0.844	0.016

(d) Uncertainty---Dice

Figure A11: The results for slice-11 of 20CA015_N133_SAX.nii.gz

Appendix B

Middle Segmentation Case

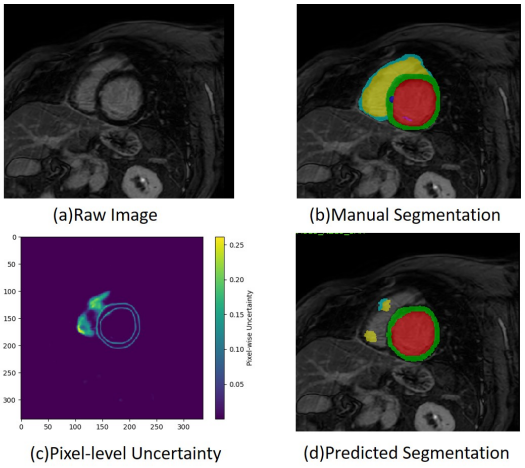
The segmentation case 20*CA*015_*N*213_*SAX*.*nii*.*gz* has the following shape: $9 \times 336 \times 336$ and its segmentation quality (measured by Dice) is 0.603.

B.1 Whole 3D Image

Table B1: The results for the whole 20CA015_N213_SAX.nii.gz Image

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.073	0.912	0.884	0.028
LV-Ep	0.225	0.724	0.696	0.028
Scar	0.438	0.392	0.433	0.041
RV-En	0.218	0.554	0.704	0.150
RV-Ep	0.464	0.228	0.399	0.171
Pap	0.413	0.412	0.463	0.051
Aor	NA	NA	NA	NA
Whole	0.262	0.603	0.651	0.048

B.2 Each Slice

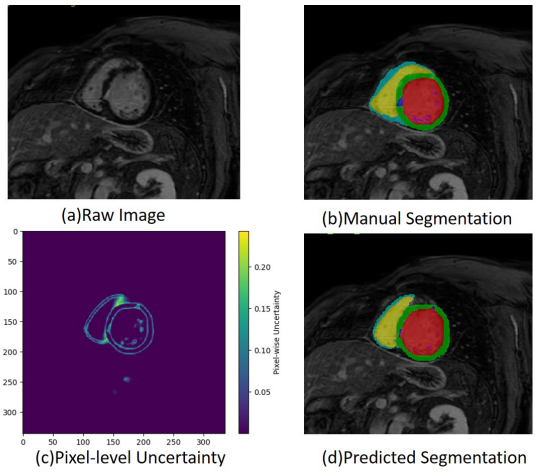


Slice 1

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.058	0.928	0.942	0.013
LV-Ep	0.182	0.731	0.81	0.079
Scar	NA	NA	NA	NA
RV-En	0.746	0.178	0.215	0.037
RV-Ep	0.867	0.006	0.087	0.08
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.265	0.406	0.723	0.317

(e) Uncertainty---Dice

Figure B1: The results for slice-1 of 20CA015_N213_SAX.nii.gz

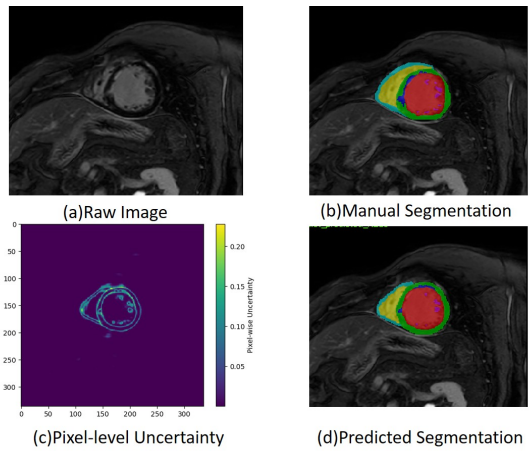


Slice 2

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.064	0.941	0.935	0.007
LV-Ep	0.172	0.834	0.822	0.013
Scar	0.516	0.345	0.458	0.113
RV-En	0.151	0.845	0.844	0.001
RV-Ep	0.321	0.505	0.664	0.158
Pap	0.508	0.282	0.466	0.183
Aor	NA	NA	NA	NA
Slice	0.247	0.679	0.741	0.062

(e) Uncertainty---Dice

Figure B2: The results for slice-2 of 20CA015_N213_SAX.nii.gz

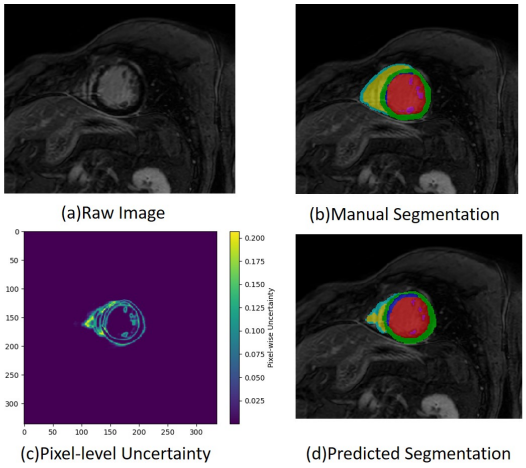


Slice 3

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.087	0.913	0.911	0.002
LV-Ep	0.182	0.821	0.81	0.011
Scar	0.403	0.283	0.577	0.294
RV-En	0.159	0.717	0.835	0.117
RV-Ep	0.334	0.249	0.651	0.401
Pap	0.454	0.586	0.523	0.063
Aor	NA	NA	NA	NA
Slice	0.231	0.653	0.758	0.105

(e) Uncertainty---Dice

Figure B3: The results for slice-3 of 20CA015_N213_SAX.nii.gz

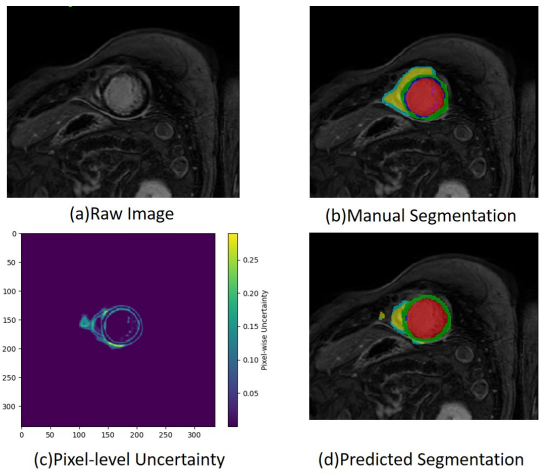


Slice 4

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.097	0.901	0.901	0.001
LV-Ep	0.197	0.777	0.795	0.017
Scar	0.352	0.322	0.632	0.309
RV-En	0.464	0.336	0.513	0.176
RV-Ep	0.548	0.010	0.424	0.413
Pap	0.458	0.650	0.52	0.131
Aor	NA	NA	NA	NA
Slice	0.302	0.571	0.684	0.112

(e) Uncertainty---Dice

Figure B4: The results for slice-4 of 20CA015_N213_SAX.nii.gz

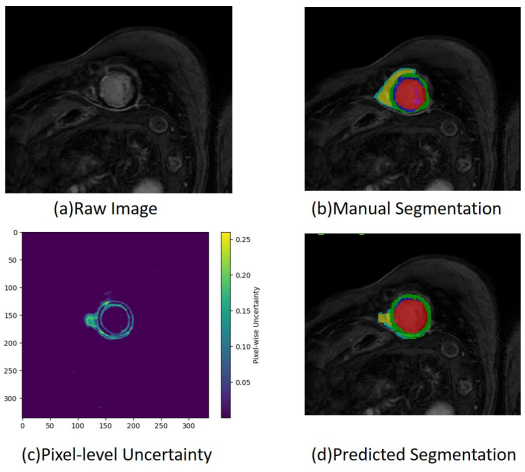


Slice 5

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.074	0.924	0.925	0
LV-Ep	0.230	0.775	0.761	0.015
Scar	0.781	0.177	0.178	0.001
RV-En	0.392	0.482	0.589	0.106
RV-Ep	0.511	0.326	0.463	0.136
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.284	0.526	0.703	0.176

(e) Uncertainty---Dice

Figure B5: The results for slice-5 of 20CA015_N213_SAX.nii.gz

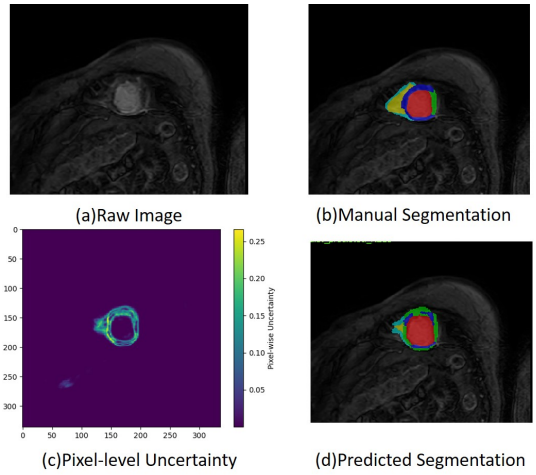


Slice 6

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.087	0.879	0.911	0.032
LV-Ep	0.251	0.718	0.737	0.019
Scar	0.603	0.359	0.366	0.006
RV-En	0.648	0.389	0.318	0.071
RV-Ep	0.669	0.137	0.297	0.16
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.323	0.497	0.662	0.164

(e) Uncertainty---Dice

Figure B6: The results for slice-6 of 20CA015_N213_SAX.nii.gz

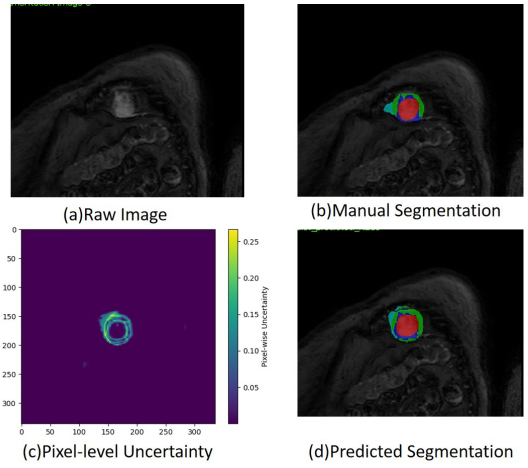


Slice 7

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.148	0.884	0.846	0.038
LV-Ep	0.492	0.367	0.483	0.115
Scar	0.477	0.420	0.499	0.078
RV-En	0.661	0.282	0.305	0.022
RV-Ep	0.704	0.228	0.259	0.031
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.355	0.597	0.628	0.030

(e) Uncertainty---Dice

Figure B7: The results for slice-7 of 20CA015_N213_SAX.nii.gz

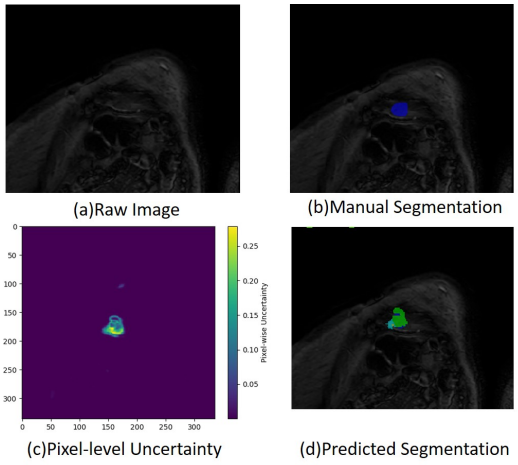


Slice 8

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.186	0.920	0.806	0.115
LV-Ep	0.440	0.642	0.538	0.104
Scar	0.591	0.659	0.379	0.28
RV-En	NA	NA	NA	NA
RV-Ep	0.754	0.060	0.207	0.146
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.281	0.754	0.706	0.048

(e) Uncertainty---Dice

Figure B8: The results for slice-8 of 20CA015_N213_SAX.nii.gz



Slice 9

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	NA	NA	NA	NA
LV-Ep	0.520	0.00	0.453	0.453
Scar	0.857	0.220	0.098	0.123
RV-En	0.900	0.00	0.052	0.052
RV-Ep	0.897	0.00	0.056	0.056
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.453	0.460	0.524	0.064

(e) Uncertainty---Dice

Figure B9: The results for slice-9 of 20CA015_N213_SAX.nii.gz

Appendix C

Poor Segmentation Case

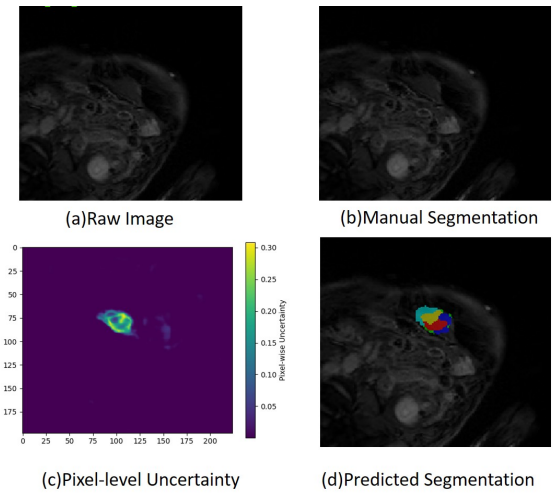
The segmentation case $20CA015\_N064\_SAX.nii.gz$ has the following shape: $12 \times 224 \times 198$ and its segmentation quality (measured by Dice) is 0.501.

C.1 Whole 3D Image

Table C1: The results for the whole 20CA015_N064_SAX.nii.gz Image

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.165	0.817	0.770	0.047
LV-Ep	0.255	0.669	0.659	0.01
Scar	0.817	0.009	0.000	0.009
RV-En	0.232	0.751	0.687	0.064
RV-Ep	0.513	0.314	0.340	0.026
Pap	0.348	0.546	0.544	0.002
Aor	0.424	0.398	0.450	0.052
Whole	0.393	0.501	0.488	0.013

C.2 Each Slice

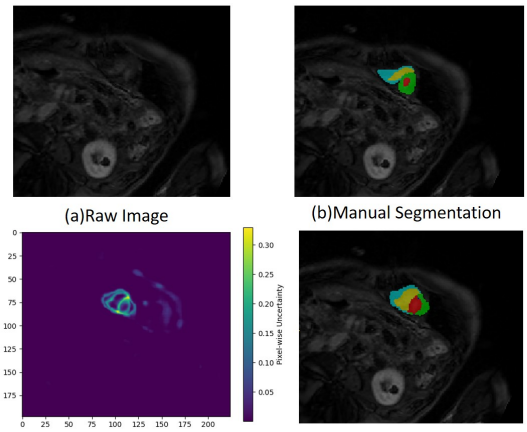


Slice 1

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.627	0.000	0.341	0.341
LV-Ep	0.950	0.000	0	0
Scar	0.872	0.000	0.082	0.082
RV-En	0.725	0.000	0.237	0.237
RV-Ep	0.694	0.000	0.27	0.27
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.552	0.285	0.4197	0.134

(e) Uncertainty---Dice

Figure C1: The results for slice-1 of 20CA015_N064_SAX.nii.gz

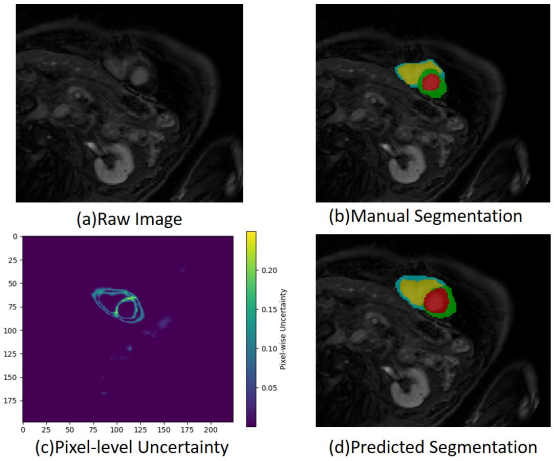


Slice 2

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.606	0.469	0.363	0.107
LV-Ep	0.530	0.462	0.444	0.019
Scar	0.872	0.00	0.082	0.082
RV-En	0.422	0.605	0.557	0.048
RV-Ep	0.503	0.444	0.472	0.027
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.419	0.568	0.560	0.008

(e) Uncertainty---Dice

Figure C2: The results for slice-2 of 20CA015_N064_SAX.nii.gz

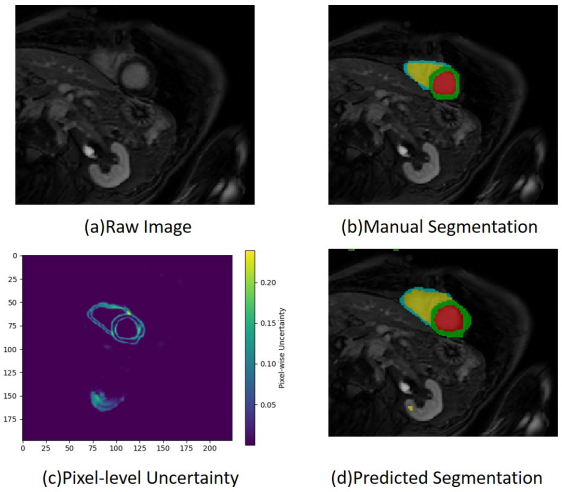


Slice 3

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.203	0.712	0.789	0.076
LV-Ep	0.305	0.435	0.681	0.246
Scar	NA	NA	NA	NA
RV-En	0.183	0.833	0.809	-0.024
RV-Ep	0.369	0.352	0.613	0.26
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.151	0.762	0.843	0.081

(e) Uncertainty---Dice

Figure C3: The results for slice-3 of 20CA015_N064_SAX.nii.gz

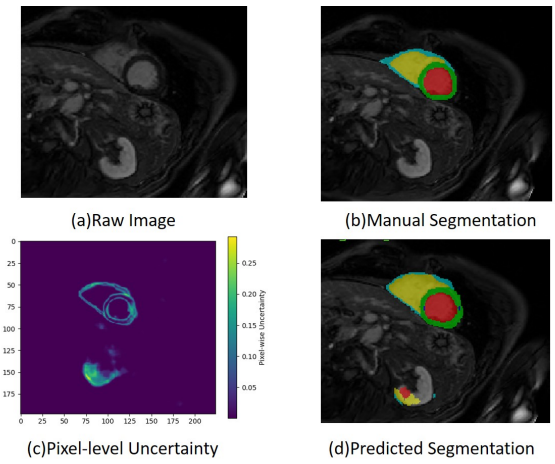


Slice 4

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.111	0.869	0.886	0.017
LV-Ep	0.224	0.564	0.767	0.202
Scar	NA	NA	NA	NA
RV-En	0.136	0.711	0.86	0.148
RV-Ep	0.363	0.212	0.62	0.407
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.119	0.765	0.877	0.112

(e) Uncertainty---Dice

Figure C4: The results for slice-4 of 20CA015_N064_SAX.nii.gz

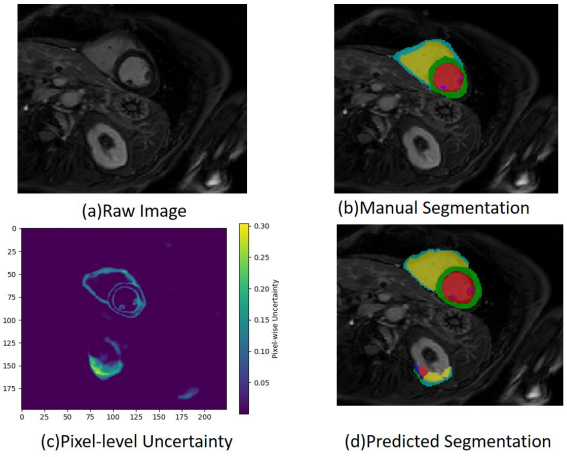


Slice 5

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.182	0.936	0.811	0.126
LV-Ep	0.274	0.759	0.714	0.045
Scar	0.895	0.000	0.058	0.058
RV-En	0.175	0.845	0.818	0.027
RV-Ep	0.498	0.344	0.477	0.133
Pap	NA	NA	NA	NA
Aor	NA	NA	NA	NA
Slice	0.289	0.697	0.698	0.001

(e) Uncertainty---Dice

Figure C5: The results for slice-5 of 20CA015_N064_SAX.nii.gz

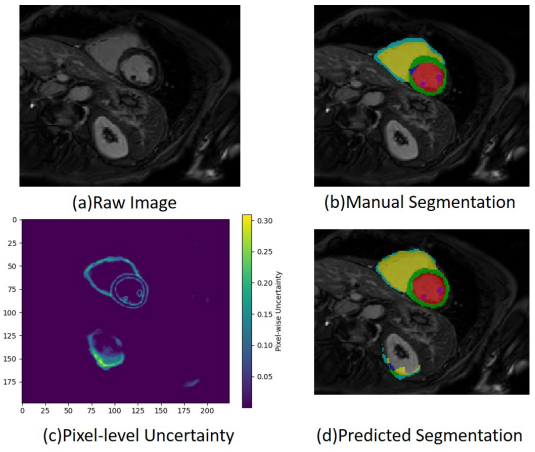


Slice 6

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.145	0.899	0.85	0.05
LV-Ep	0.216	0.738	0.774	0.036
Scar	0.950	0.000	0	0
RV-En	0.228	0.822	0.762	0.06
RV-Ep	0.531	0.299	0.442	0.143
Pap	0.515	0.528	0.459	0.07
Aor	NA	NA	NA	NA
Slice	0.369	0.612	0.613	0.001

(e) Uncertainty---Dice

Figure C6: The results for slice-6 of 20CA015_N064_SAX.nii.gz

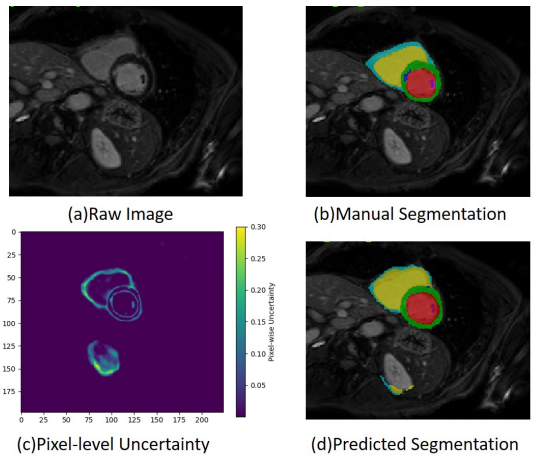


Slice 7

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.101	0.902	0.896	0.006
LV-Ep	0.222	0.753	0.768	0.015
Scar	0.984	0.000	-0.036	0.036
RV-En	0.200	0.836	0.792	0.045
RV-Ep	0.610	0.293	0.358	0.064
Pap	0.279	0.750	0.709	0.041
Aor	NA	NA	NA	NA
Slice	0.342	0.647	0.641	0.006

(e) Uncertainty---Dice

Figure C7: The results for slice-7 of 20CA015_N064_SAX.nii.gz

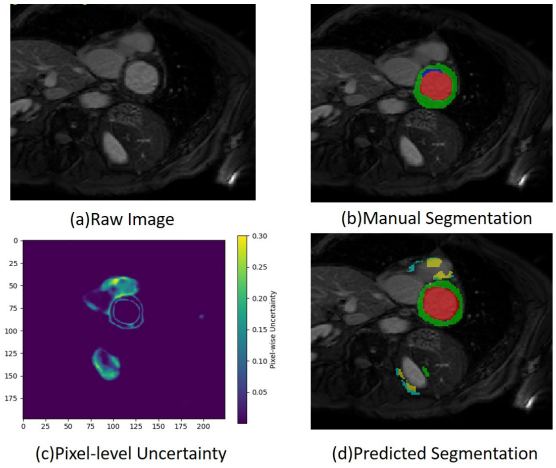


Slice 8

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.089	0.898	0.909	0.011
LV-Ep	0.180	0.791	0.813	0.021
Scar	0.743	0.000	0.218	0.218
RV-En	0.204	0.833	0.788	0.045
RV-Ep	0.715	0.356	0.247	0.109
Pap	0.804	0.285	0.154	0.132
Aor	NA	NA	NA	NA
Slice	0.391	0.595	0.590	0.005

(e) Uncertainty---Dice

Figure C8: The results for slice-8 of 20CA015_N064_SAX.nii.gz

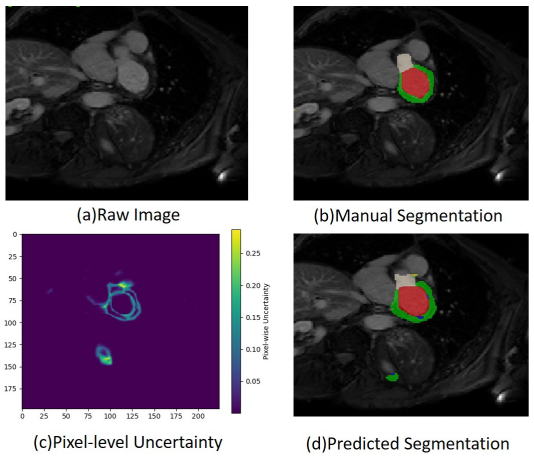


Slice 9

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.055	0.907	0.945	0.038
LV-Ep	0.269	0.774	0.719	0.056
Scar	0.758	0.000	0.203	0.203
RV-En	0.782	0.000	0.177	0.177
RV-Ep	0.886	0.000	0.067	0.067
Pap	NA	NA	NA	NA
Aor	0.800	0.000	0.158	0.158
Slice	0.507	0.383	0.467	0.084

(e) Uncertainty---Dice

Figure C9: The results for slice-9 of 20CA015_N064_SAX.nii.gz

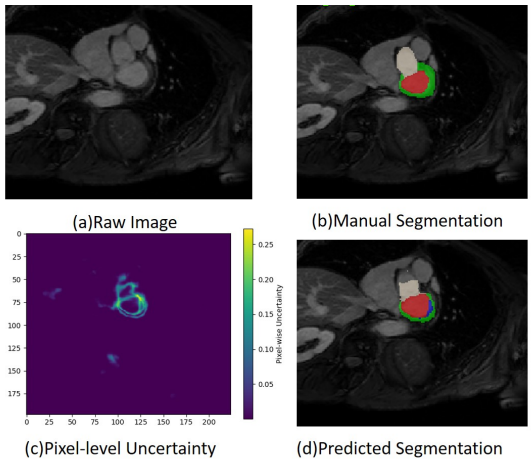


Slice 10

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.127	0.928	0.869	0.06
LV-Ep	0.360	0.626	0.623	0.003
Scar	0.767	0.000	0.192	0.192
RV-En	0.952	0.000	-0.003	0.003
RV-Ep	NA	NA	NA	NA
Pap	NA	NA	NA	NA
Aor	0.459	0.724	0.518	0.206
Slice	0.381	0.611	0.601	0.011

(e) Uncertainty---Dice

Figure C10: The results for slice-10 of 20CA015_N064_SAX.nii.gz

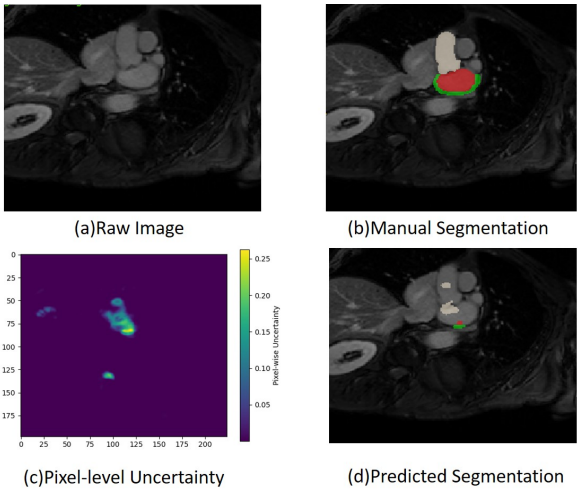


Slice 11

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.260	0.912	0.729	0.184
LV-Ep	0.520	0.549	0.454	0.095
Scar	0.692	0.000	0.272	0.272
RV-En	NA	NA	NA	NA
RV-Ep	NA	NA	NA	NA
Pap	NA	NA	NA	NA
Aor	0.359	0.691	0.623	0.068
Slice	0.261	0.736	0.7271	0.009

(e) Uncertainty---Dice

Figure C11: The results for slice-11 of 20CA015_N064_SAX.nii.gz



(a)Raw Image

(b)Manual Segmentation

(c)Pixel-level Uncertainty

(d)Predicted Segmentation

Slice 12

Class	Uncertainty	Real_Dice	Predicted_Dice	Error
LV-En	0.865	0.107	0.089	0.019
LV-Ep	0.778	0.260	0.181	0.08
Scar	0.875	0.00	0.079	0.079
RV-En	NA	NA	NA	NA
RV-Ep	NA	NA	NA	NA
Pap	NA	NA	NA	NA
Aor	0.575	0.133	0.395	0.262
Slice	0.442	0.500	0.536	0.036

(e) Uncertainty---Dice

Figure C12: The results for slice-11 of 20CA015_N064_SAX.nii.gz