Summer 6-27-2019

Brain MR Image Segmentation: From Multi-Atlas Method To Deep Learning Models

Jie Huo
University of Windsor

BRAIN MR IMAGE SEGMENTATION: FROM MULTI-ATLAS METHOD TO DEEP LEARNING MODELS

by
Jie Huo

A Dissertation
Submitted to the Faculty of Graduate Studies
through the Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada

BRAIN MR IMAGES SEGMENTATION: FROM MULTI-ATLAS METHOD TO DEEP LEARNING MODELS

by

Jie Huo

APPROVED BY:

P.Atrey, External Examiner

University at Albany, State University of New York

R. Gras

School of Computer Science

H. Wu

Department of Electrical and Computer Engineering

E. Abdel-Raheem

Department of Electrical and Computer Engineering

J. Wu, Advisor

Department of Electrical and Computer Engineering

June 27, 2019

Declaration of Co-Authorship / Previous Publication

I Co-Authorship Declaration

I hereby declare that this dissertation incorporates material that is result of joint research, as follows: This dissertation also incorporates the outcome of a research under the supervision of professor Jonathan Wu and collaboration with Dr. Guanghui Wang (Chapter 3, Chapter 4), Dr. Akilan Thangarajah (Chapter 3) and Dr. Jiuwen Cao (Chapter 4). The research under Jonathan Wu is covered in Chapter 4, 5, 6, and 7 of the dissertation. In all cases, the key ideas, primary contributions, experimental designs, data analysis, interpretation, and writing were performed by the author, and the contribution of the co-authors was primarily through the provision of proof reading and reviewing the research papers regarding the technical content.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my dissertation, and have obtained written permission from each of the co-authors to include the above materials in my dissertation.

I certify that, with the above qualification, this dissertation, and the research to which it refers, is the product of my own work.

II Previous Publication

This dissertation includes four original papers that have been previously published/under review in peer reviewed journals and conferences, as follows:

Thesis Chapter	**Publication title/full citation**	**Publication status**
Chapter 3	J. Huo, G. Wang, QM. Wu, and T. Akilan "Label fusion for multi-atlas segmentation based on majority voting." International Conference Image Analysis and Recognition, pages 100106. Springer, 2015.	Published
Chapter 4	J. Huo, QM. Wu, J. Cao, and G, Wang "Supervoxel based method for multi-atlas segmentation of brain MR images." NeuroImage, 175:201-214, 2018	Published

Chapter 5	J. Huo and QM. Wu "AttentionNet: brain anatomical structure segmentation using CNN with attention mechanism."	Under preparation for submission
Chapter 6	J. Huo and QM. Wu, "End-to-end trainable CNN-CRF with high order potentials."	Under preparation for submission

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

III General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis. I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

Quantitative analysis of the brain structures on magnetic resonance (MR) images plays a crucial role in examining brain development and abnormality, as well as in aiding the treatment planning. Although manual delineation is commonly considered as the gold standard, it suffers from the shortcomings in terms of low efficiency and inter-rater variability. Therefore, developing automatic anatomical segmentation of human brain is of importance in providing a tool for quantitative analysis (e.g., volume measurement, shape analysis, cortical surface mapping). Despite a large number of existing techniques, the automatic segmentation of brain MR images remains a challenging task due to the complexity of the brain anatomical structures and the great inter- and intra- individual variability among these anatomical structures.

To address the existing challenges, four methods are proposed in this thesis. The first work proposes a novel label fusion scheme for the multi-atlas segmentation. A two-stage majority voting scheme is developed to address the over-segmentation problem in the hippocampus segmentation of brain MR images. The second work of the thesis develops a supervoxel graphical model for the whole brain segmentation, in order to relieve the dependencies on complicated pairwise registration for the multi-atlas segmentation methods. Based on the assumption that pixels within a supervoxel are supposed to have the same label, the proposed method converts the voxel labeling problem to a supervoxel labeling problem which is solved by a maximum-a-posteriori (MAP) inference in Markov random field (MRF) defined on supervoxels. The third work incorporates attention mechanism into convolutional neural networks (CNN), aiming at learning the spatial dependencies between the shallow layers and the deep layers in CNN and producing an aggregation of the attended local feature and high-level features to obtain more precise segmentation results. The fourth method takes advantage of the success of CNN in computer vision, combines the strength of the graphical model with CNN, and integrates them into an end-to-end training network.

The proposed methods are evaluated on public MR image datasets, such as MICCAI2012, LPBA40, and IBSR. Extensive experiments demonstrate the effectiveness and superior performance of the three proposed methods compared with the other state-of-the-art methods.

Dedication

This disseration is dedicated to my beloved husband and parents whose love, encouragement and support have enriched my sour and inspired me to complete the research.

Acknowledgements

I would like to express my special appreciation and thanks to my advisor, Dr. Q.M. Jonathan Wu for giving me the opportunity to work under his supervision as well as for his guidance and continuous support for my Ph.D. study and research. Additionally, I like to thank the committee members, Dr. Robin Gras, Dr. Esam Abdel-Raheem, and Dr. Huapeng wu for taking time out of their busy schedule to come over and help me with their insightful comments and encouragement. I like to convey my sincere gratitude to Dr. Guanghui Wang, who helped me to learn the fundamental items of the machine learning domain. Furthermore, I sincerely appreciate the department graduate secretary Ms. Andria Ballo for all her support and guidance.

I sincerely thank my beloved husband, Chen, who continuously motivated me and supported me throughout my Ph.D. program. Words cannot express how grateful I am to my mother, father, my mother-in-law, and father-in-law for all of the sacrifices that they have made on my behalf. I thank my fellow labmates for the stimulating discussions and for all the fun we have had in the last few years.

Finally, I convey my sincerest regards to Google, Wikipedia and the researchers around the world, helping us to free our minds, grow our knowledge and come out of the darkness of ignorance, false beliefs and judgments. They have helped me believe that this gradual progress will lead to self-awareness and help us make a better world.

Table of Contents

List of Tables

List of Figures

List of abbreviation

2D	Two-Dimension or Two-Dimensional
3D	Three-Dimension or Three-Dimensional
BN	Batch Normalization
CNN or ConvNet	Convolutional Neural Network
CRF	Conditional Random Field
EM	ExpectationMaximization
FCN	Fully Convolutional Network
JLF	Joint Label Fusion
LMFB	Leung-Malik Filter Banks
MAP	Maximum-A-Posteriori
MRF	Markov Random Field
MRI	Magnetic Resonance Imaging
MR	Magnetic Resonance
MV	Majority Voting
PB	Patch-Based
RBF	Radial Basis Function
ReLU	Rectified Linear Unit
RDA	Regularized Dual Averaging
ROI	Region Of Interest
SLIC	Simple Linear Iterative Clustering
SVM	Support Vector Machine
SVMAF	SVM Segmentation with Augmented Features
SSD	Summed Squared Distance
STAPLE	Simultaneous Truth and Performance Level Estimation
TRW-S	Tree-Reweighted Message Passing

Chapter 1

Introduction

In this chapter, we start with introducing the topic of this Ph.D. dissertation — anatomical structure segmentation of the brain MR images in Section 1.1. Next, the motivation of the study and the research challenges related to the brain anatomical segmentation are presented in Section 1.2 and Section 1.3, respectively. Then, we clarify the contributions of the work in Section 1.4. Finally, the structure of this dissertation is explained in Section 1.5.

1.1 Segmentation of Brain MR Images

As an essential task in medical image analysis, segmentation aims at providing each pixel/voxel a label which refers to the tissue or the anatomical structure. The segmentation result is either a set of contours describing the region boundaries or an image of labels which identifies each homogeneous region [29]. The brain segmentation problem discussed in this thesis mainly focuses on brain anatomical structure segmentation, which relates to assigning each pixel/voxel in the image with a label associated with an anatomical structure in the brain.

Brain anatomical segmentation plays an important role in clinical applications. Growing evidence has shown that neurological disorders such as multiple sclerosis [17, 74, 89], stroke [59, 78], and Alzheimer's disease [30, 57] are associated with struc-

tural changes in the brain, resulting in volume or shape alternations in magnetic resonance (MR) images. Accurate brain anatomical segmentation is widely used to study the morphometric changes or to measure the volume for characterizing the neurological disorders. Moreover, segmentation not only contributes to examine the brain development and abnormality but also plays an important role in detection and localization of the abnormal tissues and surrounding healthy structures, which is an essential task for surgical planning, postoperative analysis, and chemo/radiotherapy planning [2, 54]. In addition, the segmented brain usually serves as the preliminary step of many brain image analysis, such as cortical surface mapping [38] and brain images registration and warping [96], of which the performance directly influences the outcome of following procedures. Except for the clinical applications, the anatomically segmented brain also provides a framework of functional visualization and quantitive analysis for studying and analyzing the abnormalities such as neurodegenerative disorder, psychiatric disorders, and healthy aging.

Magnetic resonance imaging (MRI) is an imaging technology that produces three-dimensional detailed anatomical images. Since the MRI offers high-resolution images and shows high contrast between soft tissues, MRI becomes the most popular medical imaging modality used for quantitive and qualitative analysis of the brain structures. Therefore, anatomical segmentation on brain MR images provides an effective tool for the anatomical and functional study of the brain.

1.2 Motivation

Because of the crucial role that segmented brain MR images play in research and clinical applications, precise anatomical segmentation becomes an essential prerequisite for the quantitative assessment of the brain.

Traditionally, the brain segmentation on MR images is accomplished by trained experts. The manual delineation (sometimes called "annotation") is usually consid-

ered as the "gold standard". However, the manual annotation can take up to a week for high-resolution MR images [35]. Moreover, it suffers from the shortcoming of intra- and inter-rater variability [24]. As a result, the manual segmentation is prone to errors and difficult to reproduce. Therefore, the manual delineation is not suitable for deploying on large-scale datasets or in applications where time is critical [50].

On the other hand, some fully-automated algorithms, e.g., thresholding, region growing, clustering methods, yield high accuracy in specific problems. Generally, the fully-automated algorithms rely on the intensity information of the MR images to classify the pixel/voxel or utilize a probabilistic atlas which stores the spatial information to aid the intensity-based segmentation. Unfortunately, these fully-automated algorithms only work for some specific segmentation tasks, e.g., tissue classification of gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF) [35], while they are not applicable to the detailed segmentation due to the overlap of intensity profiles for the complicated anatomical structures of the brain. Moreover, the performance of the fully-automated methods is limited by the artifacts in the MR images, including intensity inhomogeneity, noise, and partial volume [29].

As a middle-ground method between manual delineation and fully-automated method, the supervised methods learn/encode the relationship between the labels and the intensity images from the manually annotated training data, and predict the optimal segmentation for the target unlabeled image. Most importantly, the supervised methods can deal with the segmentation of the complicated anatomical structures. Inspired by the recent success in brain segmentation achieved by the supervised methods, e.g., atlas-based methods and learning-based methods, this thesis focuses on developing supervised segmentation algorithm for whole brain segmentation to produce fast, repeatable and accurate results.

1.3 Challenges

Brain anatomical segmentation is a challenging task despite significant efforts made by scientists and researchers. Figure 1.1 depicts some 2D slices of the brain images and label images, which reflects the challenges in brain anatomical segmentation. In this section, we present several research challenges for the field of brain anatomical structure segmentation.

Intensity overlap. MR signal holds the properties to differentiate brain and nonbrain tissues or even to distinguish among GM, WM, and CSF. Some methods thus achieve competitive performance [34, 102] in basic tissue classification (i.e., GM, WM, and CSF) or brain extraction (i.e., brain and non-brain tissue). However, as shown in Figure 1.1, the intensity overlap between distributions of different anatomical structures is severe, especially in the cortical area.

Large variations in shape, size, appearance. The shape, volume, and appearance of the anatomical structures relate to the gender, age, and pathological conditions with tumors, lesions, and edemas. These factors result in significant intra-class variations among different subjects. For example, as shown in Figure 1.1, there are considerable variations in the appearance, shape, and the size of the brain anatomical structures among different subjects. Moreover, the quality of the MR images is also affected by the scanner, machine, and even acquisition time.

Complicated labeling protocol. Compared with the basic tissue classification problem which only has three classes, the labeling protocol for anatomical segmentation is more complicated. For some datasets, the total number of classes can be more than 100. Some structures, e.g., hippocampus, are of small volume but significant pathological and physiological meaning. Therefore, successful and accurate segmentation of those small anatomical structures is a challenging task. Moreover, the inter-class variations in terms of the structure volume are quite significant for the brain anatomical segmentation, resulting in imbalanced class distribution, which

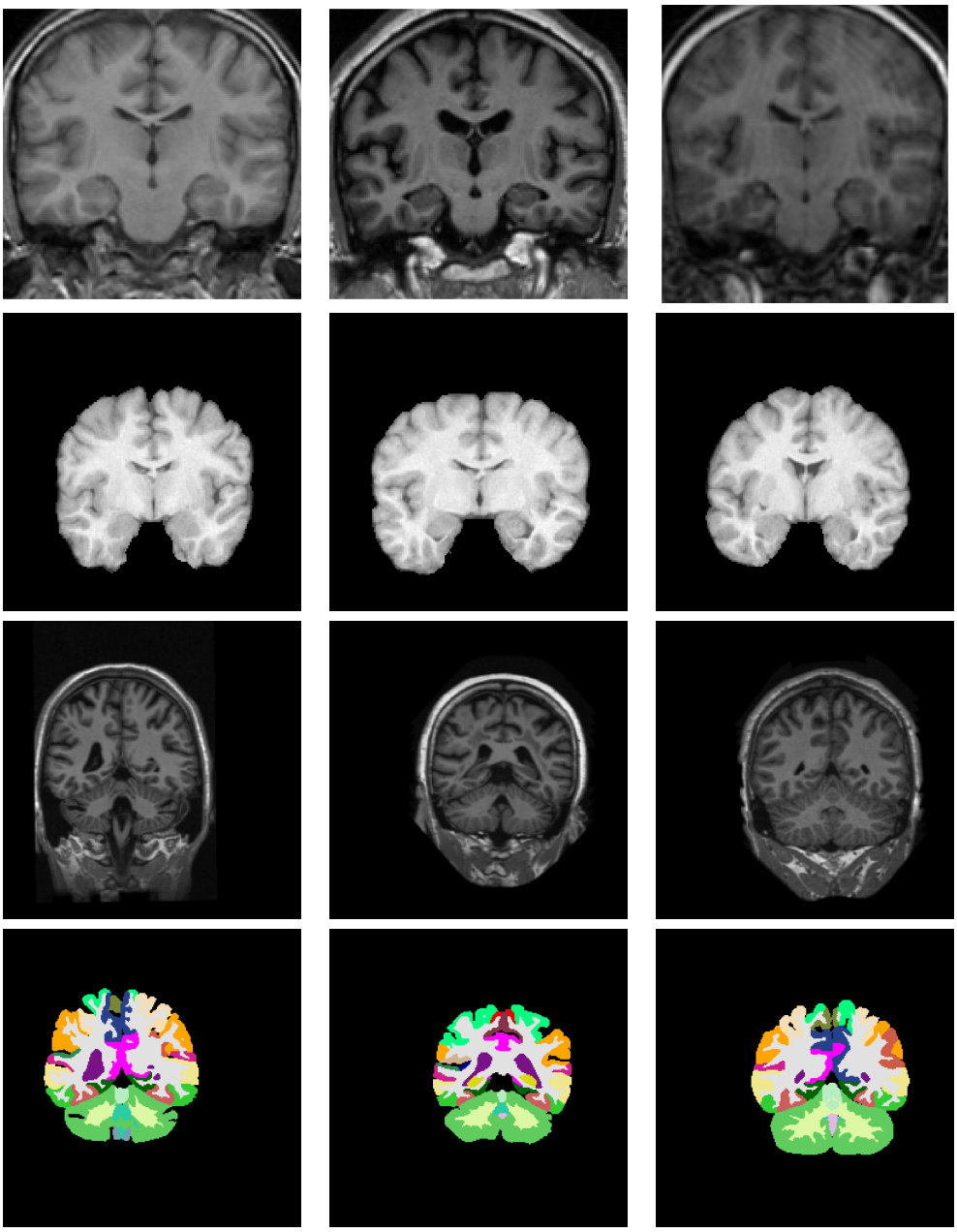


Figure 1.1: 2D slice examples of the MR images for brain anatomical segmentation. 1st row, 2nd row, and 3rd row are the intensity images from three datasets; the last row shows the corresponding label images of the 3rd row.

becomes an obstacle for some learning-based methods.

Spatial and contextual information. The spatial information plays an essential role in brain anatomy. Given a position in the brain, the number of the possible classes for the voxel is very limited. As a result, involving the position prior is a crucial point for successful segmentation of the brain anatomical structure. Moreover, the relative positions of the brain anatomical structures are fixed, e.g., the amygdala is anterior and superior to the hippocampus; structures of the left hemisphere and right hemisphere are rarely in the adjacent regions. Therefore, learning or interpreting this contextual relation also contributes to improving the segmentation performance.

Precise boundaries. Limited by the confounding appearance and the complicated labeling protocol, it is rather challenging to obtain a precise boundary for each anatomical structure. Moreover, the training data is usually too small to cover the various pattern of structure appearance. Therefore, the detailed prediction is the most challenging task for the brain anatomical segmentation problem.

Labeling inconsistency. As for the anatomical segmentation, the labeling within the neighborhood should be homogeneous. However, the labeling inconsistency problem is common in the segmentation results produced by some learning-based methods, e.g., support vector machine (SVM) and random forest. For the deep learning based methods, e.g., convolutional neural network (CNN), the labeling inconsistency is alleviated, however, the segmentation is usually followed by a graphical model, e.g., Markov random field (MRF) and conditional random fields (CRF), to refine the inconsistent labeling results.

1.4 Objective and Contributions

The objective of this Ph.D. dissertation is to develop supervised methods for improving the performance of the anatomical segmentation of brain MR image. To this end, we proposed four methods in this thesis for the aim of obtaining accurate segmen-

tation, including a two-stage majority voting scheme, a supervoxel based graphical model, a CNN with attention mechanism, and an end-to-end trainable network which combines CNN with high order CRF. This section enlists the major contributions of this dissertation as follows:

1. We develop a novel two-stage majority voting framework for multi-atlas segmentation of hippocampus on brain MR images. The first majority voting fuses the atlas labels at the image patch level with sliding a window across the target image, followed by the second majority voting which fuses the results of the first voting for the overlapping positions. We experimentally demonstrated the effectiveness of the two-stage majority voting strategy in avoiding the over-segmentation problem by comparing with the original voting scheme.

2. We propose a supervoxel based graphical model for brain anatomical segmentation. Supervoxel is an aggregation of voxels with similar attributes. Based on the assumption that the voxels within the same supervoxel have the same label, we construct the graphical model on the supervoxels. By minimizing the energy function associated with the supervoxel based graphical model, the dense labeling of MR image is converted to the supervoxel labeling problem. Since supervoxels are considered as the nodes in the graphical model, the number of variables is much less than the graphical model defined on voxels, resulting in short inference time. Moreover, because all the voxels inside the supervoxel are assigned the same label, the labeling consistency is thus encouraged within the supervoxel.

3. We propose a spatial attention model to capture the spatial dependencies between two feature maps based on the cosine similarity. We model this spatial attention function as building layers in CNN and combine it with the encoder-decoder CNN architecture. The spatial attention block connects the high-level features from the up-sampling path and the finer features from the down-sampling path and computes an attention map that highlights the related

spatial positions in the finer feature maps. By combining the related finer features with the high-level features, the net is equipped with the ability of precise localizing and detailed boundaries prediction.

4. We develop a 2D CNN architecture, which benefits the model in terms of low memory requirement, deep architecture, and fine-tuning on the pre-trained model. In order to deal with the 3D data format in MR images, we embed the spatial position information along with the intensity images in the inputs. The incorporation of the position information not only compensates for the loss of the spatial context in the third dimension but also enables the net to train on both intensity and spatial prior.

5. We propose a unified framework which combines the strength of CNN with high order CRF. Considering the characteristic of brain anatomical structures, we propose a semi-densely connected pairwise potential which encourages the smoothness of the labeling between two pixels within a neighborhood. In addition, we apply a class-specific kernel weight to the high order potential. We derive the mean field approximation for the high order CRF and model the inference as building blocks of CNN so that the CNN and high order CRF can be trained in an end-to-end fashion where the parameters are learned jointly during the training phase. By employing the superpixel based high order term, the proposed high order CRF encourages the labeling consistency among the pixels within the same superpixel. Extensive experiments demonstrate that involving the high order potential contributes to improving the segmentation accuracy compared with the other graphical models.

1.5 Organization of Thesis

The rest of the thesis is structured as follows: Chapter 2 reviews the existing supervised methods for brain anatomical segmentation. The related background details regarding the techniques used in this thesis are also included. Chapter 3 proposes

a novel two-stage majority voting scheme for multi-atlas based segmentation of hippocampus in brain MR images. Chapter 4 proposes a new whole brain segmentation framework based on supervoxel based graphical model. Chapter 5 develops an encoder-decoder CNN architecture for brain anatomical segmentation, which employs the attention mechanism to improve the ability of detailed prediction. Chapter 6 develops an end-to-end training network which integrates the CNN with the high order CRF for whole brain segmentation on MR images. Chapter 7 concludes the thesis with overall discussions and intuitive directions for future work.

Chapter 2

Background

2.1 Overview

For anatomical segmentation of the brain MR images, the existing methodologies can be categorized into three groups: multi-atlas segmentation, graphical model, and learning based method. In this chapter, we review the related works regarding the three methods along with the corresponding background knowledge.

2.2 Multi-atlases Segmentation

2.2.1 Background

Atlas-guided segmentation is a widely used method for the neuroanatomical structure segmentation. By registering the target image to the manually labeled image, one can obtain a mapping between two coordinate systems which can be used to transfer the labels from the atlas to the target image. This technique refers to the classic single-atlas segmentation procedure. However, the single atlas is not capable of dealing with the wide anatomical variation. Consequently, instead of the single atlas, multiple atlases are employed for brain anatomical structure segmentation.

Multi-atlases segmentation can be viewed as a supervised training algorithm which

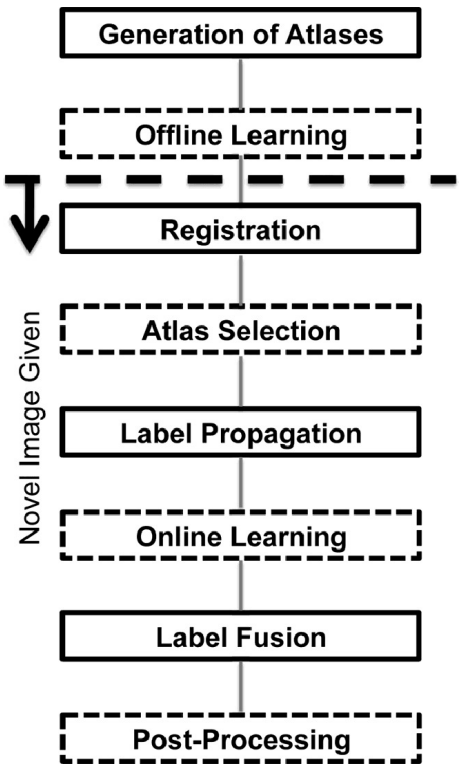


Figure 2.1: Building blocks of multi-atlas segmentation [50] (Dashed blocks are optional steps).

relies on the manually delineated data (commonly called "atlas"). In this approach, each atlas is potentially used for segmenting the target image. A typical multi-atlases segmentation method includes applying registration between the target image and each atlas image (commonly called "pairwise registration"). The pairwise registration establishes the voxelwise spatial correspondence between the target and each atlas image. Based on the registration results, the atlas labels are transformed to the target image space, which refers to "label propagation". Then, by fusing those candidate labels, the optimal label is obtained at each voxel, which is called "label fusion". Figure 2.1 depicts a general framework of the multi-atlas segmentation method. In the rest of the section, we review two key components in Figure 2.1, pairwise registration and label fusion, which relate to the work of this thesis.

2.2.2 Related Work

Registration. In multi-atlas segmentation, registration is in charge of establishing the spatial correspondence between the target image and each atlas. Based on the geometric transformation, it can be divided into rigid and non-rigid registration. Rigid registration is usually applied to rigid structures, e.g., bones, or employed as a pre-registration strategy. For brain anatomical structures, the multi-atlas segmentation methods usually adopt complex deformable models which assign each location a spatial transformation vector, such as nonlinear deformable models [42, 86] or non-parametric diffeomorphisms [15, 110].

For multi-atlas segmentation, since the registration is performed between the target and each atlas pairwisely, the registration step becomes the computational bottleneck. Some methods can reduce the computational burden of the pairwise registration by reducing the number of atlases [124]. Alternatively, some research co-registered all the atlases to construct a template atlas. By performing the registration between the target and the template atlas, this approach can reduce the computation cost of registration but also might lead to the decrease of the performance due to the suboptimal registrations [4, 98]. Moreover, the patch-based technique searches the neighborhood in the atlas and thus relax the one-to-one correspondence assumption in multi-atlas segmentation. Therefore, patch-based methods can be combined with the multi-atlas segmentation for alleviating the requirements for high accurate pairwise registration. [8, 13, 88, 116].

Label fusion. In multi-atlases segmentation, the segmentation errors stem from the registration errors. As the core of the multi-atlases segmentation algorithm, label fusion is applied to account for the registration errors. Majority voting [48, 91], which selects the most frequent label at each position, is the simplest yet efficient label fusion strategy. In order to utilize the image intensity information, majority voting was extended to weighted fusion by assigning each atlas a global/local weight based on the similarity between the target and the atlas. The global weighted fusion

utilizes the global information and associate each atlas with a unique weight which is estimated by comparing the mutual information [6] or by posing it as a least square problem [18]. However, the global weight cannot explain the spatial variety. Instead, local weighted fusion methods are developed to use local similarities between the atlas and the target (e.g., local absolute difference [55], local cross-correlation [7], Gaussian intensity difference function[58], and Jacobian determinant of the deformation fields [85]). Wang et al. [117] developed the joint label fusion to account for the correlations of label errors produced by different atlases in the voting strategy. The weights are optimized to minimize the total expected segmentation error, which relates to the pairwise dependencies among the atlases.

Sabuncu et al. [91] proposed a generative model for label fusion, which is formulated by marginalizing the conditional probability with respect to a mapping field. By configuring the mapping field, the model evolves into different label fusion algorithms and generalize the global and local weighted fusion methods. Based on this generative probabilistic model, Iglesias et al. [51] applied a joint histogram instead of the Gaussian noise in [91], extending the generative model to intermodality fusion; Bai et al. [13] integrated the patch-based method with the generative model, leading to a probabilistic patch-based label fusion.

In addition, another category of probabilistic label fusion methods is established on the simultaneous truth and performance level estimation (STAPLE) [120], which integrates a stochastic model of rater behavior into the estimation process. Many works have been developed in modifying the original probabilistic model, including defining data-driven a priori distribution [70], introducing a hierarchical noise model [9], and integrating non-local correspondence into the STAPLE framework [8].

Another progress in multi-atlas segmentation is the application of the patch-based technique, which is derived from image denoising [26, 62]. Based on the intuitive idea that similar patches tend to have the same label, patch-based technique searches for similar patches in the neighborhood of the atlas images. As done in [13, 25, 117], the

patch-based technique can be incorporated into the label fusion scheme and account for the registration errors.

2.3 Random Field for Segmentation Problem

2.3.1 Background

The segmentation problem can be posed as a MAP estimation for an appropriately defined graphical model which is associated with a CRF [65, 68, 99]. Given an image I, a random field is defined over a set of random variables $\mathbf{X} = \{X_1, X_2, \ldots, X_N\}$, where each random variable is associated with a corresponding image pixel $i \in \{1, 2, \ldots, N\}$ and takes a value from the label set $\mathcal{L} = \{l_1, l_2 \ldots, l_k\}$. The CRF (X, I) is characterized by a Gibbs distribution:

$$P(\mathbf{X} \mid \mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\Big(- \sum_{c \in \mathcal{C}} \psi_c(\mathbf{X}_c \mid \mathbf{I}) \Big) \tag{2.1}$$

where the partition function $Z(\mathbf{I})$ is a normalizing constant, clique c is a set of random variables that are conditionally dependent on each other, $\mathcal{C}$ is the set of all the cliques, and ψ_c is the potential term induced by the clique c. The Gibbs energy of the labeling configuration $\mathbf{x} \in \mathcal{L}^N$ is:

$$E(\mathbf{x}) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c) \tag{2.2}$$

where notation of conditioning on $\mathbf{I}$ is omitted for convenience. Therefore, the MAP labeling of the random field in Equation (2.1) corresponds to minimizing the Gibbs energy function in Equation (2.2).

Based on the definition of the cliques and the corresponding potentials, the CRF model can be divided into three models:

Adjacency CRF: In the adjacency CRF model, the clique set $\mathcal{C}$ involves the unary cliques and the pairwise cliques. The energy function is:

$$E(x) = \sum_{i \in \mathcal{V}} \psi_u(x_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_p(x_i, x_j) \tag{2.3}$$

Each unary clique is associated with a random variable X_i, the corresponding unary potential ψ_u measures the cost of assigning label x_i to pixel i. The pairwise clique consists of a pair of random variables, X_i and its neighbor X_j. The pairwise potential ψ_p measures the cost of assigning label x_i and x_j to pixel i and j simultaneously.

Fully connected CRF: The only difference between adjacency CRF and fully connected CRF is that the fully connected potentials are defined over all the pixel pairs in the image instead of a neighborhood system. The energy function is:

$$E(x) = \sum_{i \in \mathcal{V}} \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, x_j) \tag{2.4}$$

High order CRF: Besides unary cliques and pairwise cliques, the high order CRF involves the high order clique which refers to a set of variables $X_s = \{X_1, X_2, \ldots, X_M\}$. The high order CRF is of the form:

$$E(x) = \sum_{i \in \mathcal{V}} \psi_u(x_i) + \sum_{i,j} \psi_p(x_i, x_j) + \sum_{c \in \mathcal{S}} \psi_h(x_c) \tag{2.5}$$

where the high order potential ψ_h measures the cost of the label configuration x_s for the set of variables and $\mathcal{S}$ denotes the set of all the high order cliques. The pairwise potentials can be defined over the neighborhood system or the whole image.

2.3.2 Related Work

The graphical model has been successfully applied to brain anatomical structure segmentation [93, 108]. By employing a graphical model defined on voxels, one can obtain the optimal label for each voxel by minimizing the corresponding energy function. In these approaches, the prior knowledge is usually obtained by registering the target to a fixed probabilistic atlas[1]. Those approaches can be categorized as fully-automated

[1] probabilistic atlas is an anatomical template that retains quantitative information on inter-subject variations in brain architecture. A digital probabilistic atlas of the human brain, incorporating precise statistical information on positional variability of important functional and anatomic interfaces, may rectify many current aliasing problems since it specifically stores information on the population variability, e.g., ICBM152, MNI152, and Harvard-Oxford cortical and subcortical structural atlases.

algorithms, which are not in the scope of review in this thesis. In this section, we focus on the methods that combine the graphical model with the multi-atlas segmentation.

Regarding combining the graphical model with the multi-atlas segmentation, some methods employ the graphical model as a post-processing step following the label fusion to refine the labeling results [107]. With employing the probabilistic map obtained from the label fusion step as the spatial prior, the corresponding energy can be minimized by using graph-cuts [76, 121], or max-flow [64, 84].

Alternatively, the graphical model can serve as the registration between the target and the atlas, which turns to minimize the energy function with respect to the displacement vector [36, 43]. Moreover, some works integrate the registration and segmentation in one graphical model to solve the registration and segmentation simultaneously. Alchatzidis et al. [3] designed the MRF energy function comprising registration term and segmentation term and optimized it through dual decomposition algorithm. Gass et al. [37] cast the simultaneous segmentation and registration problem as a two-layer graph defined on MRF and developed hierarchical implementation, allowing coarse-to-fine registration and pixelwise label estimation.

Instead of using the voxel-based graphical model, some studies employ the graphical models defined on supervoxels. There are two existing ways of applying supervoxel-based graphical model to the brain segmentation problem. 1) The supervoxel-based graphical model serves as the pairwise registration, where the objective is to estimate the displacement vectors mapping the target supervoxels to the atlas supervoxels [44, 119, 128]. The assumption that no deformation exists within the supervoxel (i.e., all the voxels within the supervoxel obtain the same displacement vector) exempts the framework from the requirement for accurate supervoxel segmentation. However, the graphical inference needs to be performed by N times (N is the number of the atlas number), and following the supervoxel graphical model, a label fusion is required to fuse the candidate labels of the atlases. 2) Supervoxel graphical model is directly used in image segmentation [53]. The assumption of this framework is that all the voxels

within the supervoxel have the same label so that accurate supervoxel segmentation is a prerequisite. However, this requirement is difficult to achieve due to the intensity overlap in different anatomical structures and the lack of visible boundaries in some ROIs, especially in the cortical area. As a result, the supervoxel graphical model is usually applied to the tumor segmentation [53] or subcortical area segmentation instead of cortical structure segmentation.

High order potential has demonstrated the effectiveness in computer vision [65, 90, 111, 130]. However, it is unfeasible to compute the general higher-order potential defined over many variables [79], especially for the 3D MR data. In the medical image segmentation field, a few of studies have been proposed to involve the high order potential for encouraging the regional labeling consistency [60], encoding the shape prior [114], or embedding the boundary prior [5].

2.4 Convolutional Neural Network

Brain anatomical segmentation can be viewed as a voxel labeling problem, which makes it possible to employ a classifier (e.g., SVM classifier [12], random forest [132]) to classify the voxel. However, these traditional classifier relies on domain-specific hand-crafted features so that it is difficult to generalize the algorithm among images obtained from different modalities. Recently, due to the success of deep learning in computer vision, increasing deep networks have been developed for medical image segmentation. In this section, the basic knowledge of CNN is first introduced and follows it with the related works of using CNN for brain anatomical segmentation.

2.4.1 Background

CNN is a typical neural network which was originally used for image classification. It takes the raw image as the inputs and outputs the score of each class by stacking multiple convolution layers and fully connected layers. The whole network can be

seen as a single differentiable, parameterized function that maps the raw image to class scores. By setting the appropriate loss function, one can update the parameters based on the partial derivatives which are computed through back-propagation.

In order to build CNN, we use the following building blocks in the studies of this thesis.

Convolution layers. Convolutional layers are the core building blocks of the CNN architecture, which serve as the feature extractors that map data to the transformed feature space. It extracts the features by convolving a kernel (or filter) with the inputs. The outputs are usually called feature maps or activation maps. For the convolutional layers, there are three parameters: (1) Kernel size, which refers to the height and weight of the kernel and relates to the receptive field, which is the region of the particular output feature sees from its input space. (2) The depth, which corresponds to the number of filters that are used for the convolution operation. (3) The stride, which refers to the number of pixels that we jump when sliding the convolution kernel over the inputs. Stride greater than one results in reducing the spatial size.

Fully convolutional layer. To obtain a dense prediction using CNN, we apply the fully convolutional layer at the top of the architecture. The fully convolutional layer is 1×1 convolution with a depth of L, where L is the number of classes. By appending a fully convolutional layer at the top, one can obtain the class scores at each position simultaneously. However, due to the downsampling operations (e.g., strided convolution or pooling operation), directly applying the fully convolutional layer results in a coarse prediction. Therefore, prior to fully convolution layer, upsampling operations (e.g., deconvolution layer or interpolation) are usually adopted for applying CNN to segmentation problem.

Deconvolution layer. Deconvolution layer is the transpose of the convolution layer. Therefore, performing a deconvolution with stride greater than one leads to the increase of the spatial size of the feature maps. The rest parameters are the same as those of the convolution layer.

ReLU. As an activation function, ReLU helps a model account for interaction effects and produce the non-linear mapping. The ReLU function has a derivative of 0 for negative inputs while a derivative of 1 for positive inputs, which effectively avoids the vanishing gradient problem.

Batch Normalization. Batch normalization [52] normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. The batch normalization reduces the internal covariate shift that is produced by distribution variation of the layer's inputs during the training stage, resulting in accelerating the training speed. Moreover, by applying batch normalization layers, the net is more tolerant to increased training rates and often does not require Dropout for regularization.

Softmax function. Softmax function takes a vector of K real numbers as the input and normalizes it into a probability distribution consisting of K probabilities.

$$y(z) = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_i}} \tag{2.6}$$

After applying the softmax function, each element in the vector is normalized to the interval $(0, 1)$, and the elements are sum to 1. Thus the output of the softmax function can be interpreted as the probabilities. Furthermore, the softmax function can enlarge the difference between the elements.

It is worth noting that the softmax function has the same formulation as the Gibbs distribution in Equation (2.1), where each z_i can be considered as the energies of the variable while the denominator is the partition function.

2.4.2 Related Work

Recently, because of the success of deep learning in computer vision, increasing deep networks have been developed for medical image segmentation [27, 77, 103]. The architectures fall into two categories. One approach is the voxel/pixel classification, which directly applies the CNN to the segmentation task [28, 83, 129]. As shown in

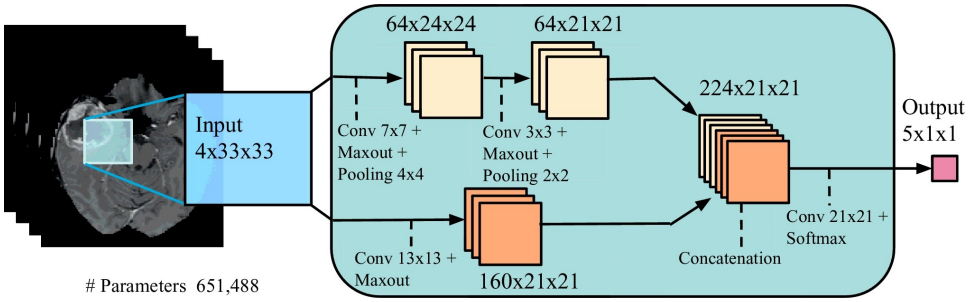


Figure 2.2: Directly applying classification net to segmentation task. [39]

Figure 2.2, the networks are trained on 2D/3D image patches cropped from the whole image and produce the prediction of the central position in the image patch. However, during the inference stage, "sliding window" is applied to obtain the pixel/voxelwise prediction, resulting in low inference speed. Moreover, the information redundancy in training patches hinders the performance of the model.

Another approach adopts the 2D/3D fully convolutional network (FCN) architecture [20, 32, 61], which replaces the fully connected layer with the fully convolutional layer. The FCN architecture takes inputs with any size and outputs the dense probability map of the input image. In order to capture both local and contextual information, architectures with parallel convolutional pathways are employed for multi-scale processing, as illustrated in Figure 2.3. Kamnitsas et al. [61] exploited a two-pathway FCN to take inputs of different resolutions. Havaei et al. [39] adopted two-pathway convolution layers with different convolutional kernel size. The multiple pathway architecture is also prevalent in the voxel/pixel classification net, e.g., in [82], the inputs are of different spatial sizes so that the net is capable of capturing multi-scale context. However, the multiple pathway networks are usually designed in a shallow fashion (two to three convolutional layers followed by fully connected/-convolutional layer) in order to take trade-off between the increasing parameters and memory required for training.

U-net [87] provides a novel way of applying CNN for segmentation. As shown in Figure 2.4, it supplements a contracting network (also called "downsampling path" or "encoder net") by an expanding network (also called "upsampling path" or "decoder

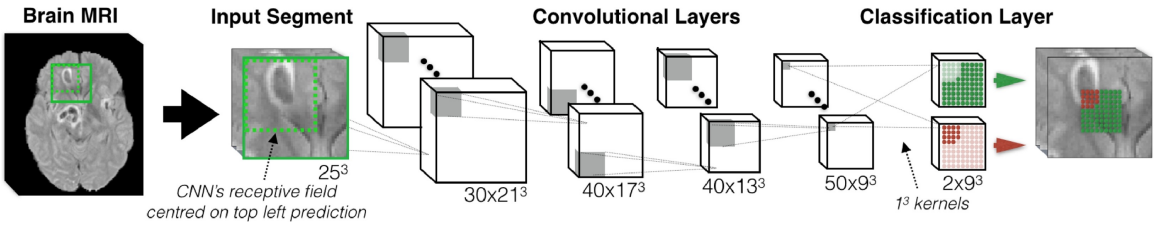


Figure 2.3: FCNN architecture with parallel convolutional pathways. [61]

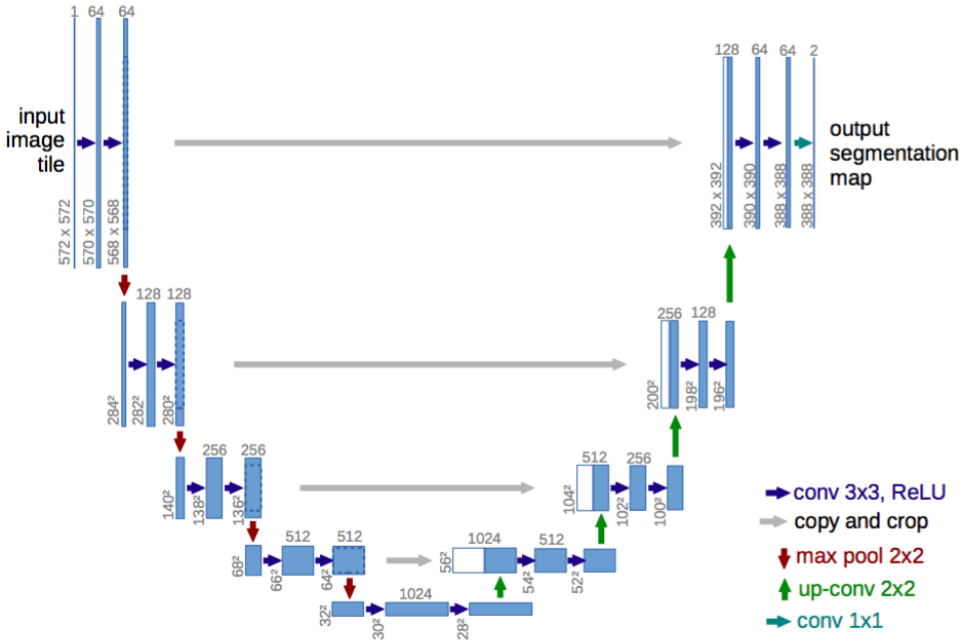


Figure 2.4: Architecture of U-net. [87]

net"). The contracting network is topologically identical to a classification net while the expanding network replaces the pooling layers with deconvolution layer to recover the resolution to the input size. The contracting net and expanding net are more or less symmetric, resulting in an u-shape in the appearance of the architecture. Furthermore, the local features from the contracting net are connected to the expanding path. The feature combination equips the U-net with the ability to capture both local and larger contextual information. Many works extend the U-net to the specific application by using different loss function [80], using different feature combining methods [127], combining with the state-of-the-art classification networks [33].

Since the spatial information provides valuable cues that represent the possible

class of a voxel, in the application of brain anatomical structure segmentation, a couple of networks encode the position information into the inputs to augment the segmentation. Wachinger et al. [112] uses the combination of spectral brain coordinates and Cartesian coordinates. de Brebisson et at. [28] adopts relative coordinates which compute the distance from voxel to the centroid of each segmentation. The aforementioned research experimentally demonstrated that the incorporation of the position information leads to improvement in the anatomical structure segmentation.

Imbalanced data among different class is a challenging task for brain segmentation even in the other medical image segmentation field. To address this problem, data resampling technique is applied to balance the sample numbers of different classes. Kamnitsas et al. [61] built training batches by cropping the segments with 50% probabilities being centered on foreground (tumor) or background (non-tumor) voxel. Havaei et al. [14, 39] adopted a two-phase training strategy to alleviate the class imbalance. In the first training phase, equally sampled training samples are used while the uniformed sampled batches are used in the second training phase with only fine-tuning the last layer. In [112], the authors employed another two-phase training strategy, which separates the brain tissues from the background in the first phase and identifies the anatomical structures in the second phase.

Furthermore, some techniques are applied to deal with the complexity in medical images. For example, integration of multiple modality images leads to significant improvement of the performance [31, 129]. Deep supervision, which adds a group of weighted auxiliary classifiers into the network, is applied by the works in [21, 22, 126] to further strengthen the training process.

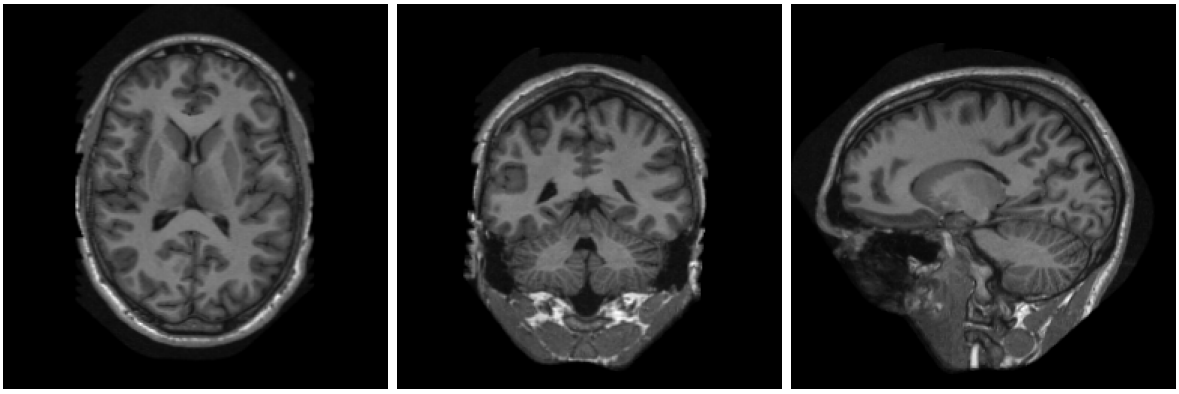
2.5 MRI Coordinate System

The anatomical space is the most important model coordinate system for medical imaging. The anatomical space is also called RAS coordinate system, where "RAS"

stands for right, anterior, and superior, respectively. With three directions left to right, posterior to anterior, and inferior to superior, this space consists of three planes to describe the standard anatomical position of a human:

1. The axial plane is parallel to the ground and separates the superior (head) from the inferior (feet).
2. The coronal plane is perpendicular to the ground and separates the anterior (front) from the posterior (back).
3. The sagittal plane separates the left from the right.

Although there is a difference between the anatomical coordinates and the 3D image coordinates, to simplify the notations in this thesis, we refer the three axises in the volumetric images space as x, y, and z, and represent three planes xy, xz, and yz as axial slice, coronal slice and sagittal slice, respectively. Figure 2.5 shows the three views of a brain MR image.



(a) Axial view (b) Coronal view (c) Sagittal view

Figure 2.5: Three planes of a brain MR image.

2.6 Datasets

There are three publicly available brain MRI datasets used in this dissertation:

1. LONI-LPBA40 dataset

   The LPBA40 dataset [97] includes 40 T1-weighted MRI scans of healthy volunteers, which is acquired on a GE 1.5T system. The dataset consists of 20 males and 20 females, age 29.20 ± 6.30 years. The 124 coronal brain slices are 1.5 mm apart with in-plane voxel resolution of 0.86 mm (38 subjects) or 0.78 mm (2 subjects). The brain is manually delineated into 50 cortical structures, 4 subcortical areas, the brainstem, and the cerebellum.

2. MICCAI 2012 Multi-Atlas Labeling Challenge dataset

   The MICCAI 2012 dataset [69] includes 35 T1-weighted MRI scans obtained from the OASIS project, where 15 subjects (5 males, 10 females, age 23.00 ± 4.12 years) are used as the atlases and the remaining (8 males, 12 females, age 40.40 ± 22.43 years) are used for testing. The labeling protocol for OASIS project is a brain labeling protocol using 134 labels, including 36 subcortical labels and 98 cortical labels.

3. IBSR dataset

   The IBSR dataset consists of 18 T1-weighted MRI scans (14 males, 4 females), provided by the Center for Morphometric Analysis at Massachusetts General Hospital and are available at http://www.cma.mgh.harvard.edu/ibsr/. Coronal slices are 1.5 mm apart with in-plane resolution of 0.9375 mm (8 subjects), 1 mm (6 subjects), or 0.8371 mm (4 subjects). The IBSR dataset consists of two types of manual segmentation: 1) the images are manually segmented into 32 subcortical structures, and 2) the cortex area is sub-divided into 96 cortical structures.

2.7 Image Pre-Processing

Image pre-processing plays a critical role in brain MR image segmentation. The pre-processing techniques used in this thesis context include:

1. Bias field correction

   MR images often exhibit image intensity inhomogeneity that is the result of magnetic field variations rather than anatomical differences. These artifacts are often described as bias, inhomogeneity, illumination non-uniformity, or gain field, can be produced by imaging instrumentation, such as radio-frequency non-uniformity, and static field inhomogeneity [16, 46]. These variations are often seen as a slowly gained signal that varies spatially. Numerous methods have been proposed to correct this artifact. In this thesis context, the N4 bias field correction [106] is applied to correct the intensity inhomogeneity.

2. Pairwise registration

   In this thesis context, the term "registration" means to determine the spatial alignment between two images of different subjects, acquired from the same dataset. Registration relates to a transformation that can associate the position of features in one image or coordinate space with the position of the corresponding feature in another image or coordinate space [45].

   In the multi-atlas segmentation, both the intensity images and the label images of the atlases are required to be warped to the target domain with the same transformation. Consequently, the first step of the registration is to generate the transformation files based on the similarity between intensity images of the atlas and the target. Then the transformation files are applied to the atlas intensity image and the label image to generate warped atlas intensity and label image, respectively.

3. Intensity normalization

   Large variations in intensity ranges widely exist in the MRI scans, which are caused by the differences in the protocols of MRI scans, various manufacturers and scanner-models, and different time points of the same patient [94]. To deal with the intensity variations, we perform the intensity normalization to ensure each tissue to have similar intensity distributions, which has been shown importance in

both supervised method and unsupervised segmentation approaches.

Chapter 3

Label Fusion for Multi-Atlas Segmentation Based on Majority Voting

Multi-atlas based segmentation is successfully applied to medical image segmentation. Majority voting, as the simplest label fusion method in multi-atlas based segmentation, is a powerful segmentation method. In this paper, a novel majority voting-based label fusion is proposed by introducing patch-based analysis for automatic segmentation of brain MR images. The proposed approach, by comparing the similarity between patches, avoids the over-segmentation problem of majority fusion. The approach is successfully applied to the segmentation of hippocampus, and the experimental results demonstrate significant performance improvement over three state-of-the-art approaches in the literature.

3.1 Introduction

Atlas-based segmentation is based on the observation that segmentation strongly correlates with image appearance. By performing registration between the target image and the atlas, the labeled atlas image is warped to the target image space. One can use the resulting warp to map the atlas label to the coordinates of the target image. For the multi-atlas segmentation method, multiple atlases are separately

registered to the target image, and voxelwise label conflicts between the registered atlases are resolved by label fusion.

In multi-atlas segmentation, the estimated segmentation is obtained by performing label fusion on the warped atlases. Although weighted fusion and statistical fusion yield good results in segmentation of magnetic resonance (MR) image [105, 118, 122, 123], the estimation of the weight and the expectationmaximization (EM) estimation, which play important roles in weighted fusion and statistical fusion, is very computationally intensive. In contrast, majority voting, which is probably the simplest label fusion method, has been demonstrated to yield powerful segmentation results with less computation. Majority voting method, however, may yield over-segmentation since it does not utilize image intensity information. The patch-based method, which compares the similarity of intensity between patches, can be combined with majority voting multi-atlases segmentation to avoid such over-segmentation errors.

Motivated by this idea, we propose a novel label fusion method which combines majority voting with the patch-based method to achieve automatic segmentation in brain MR images. The proposed method is successfully applied to the segmentation of hippocampus. In addition, the influences of different parameters are studied empirically, and a comparison with three closely related methods is performed to demonstrate the effectiveness of the proposed approach.

3.2 Method

Consider an image $I = \{I(x)|x \in \Omega\}$, where x denotes the voxel; and $\Omega \subset \mathbb{R}^3$ denotes the lattice on which the image is defined. The goal of segmentation is to estimate a label map L associated with the image I, in which each voxel is assigned a discrete label l. The label l takes discrete values from 1 to $\mathcal{L}$ for all the possible labels for the voxels in the image. In multi-atlas segmentation, I_T is a target image and $A_1, \cdots, A_n$ are n atlases with $A_i = (I_i, S_i)$, where I_i is the atlas image which has aligned to

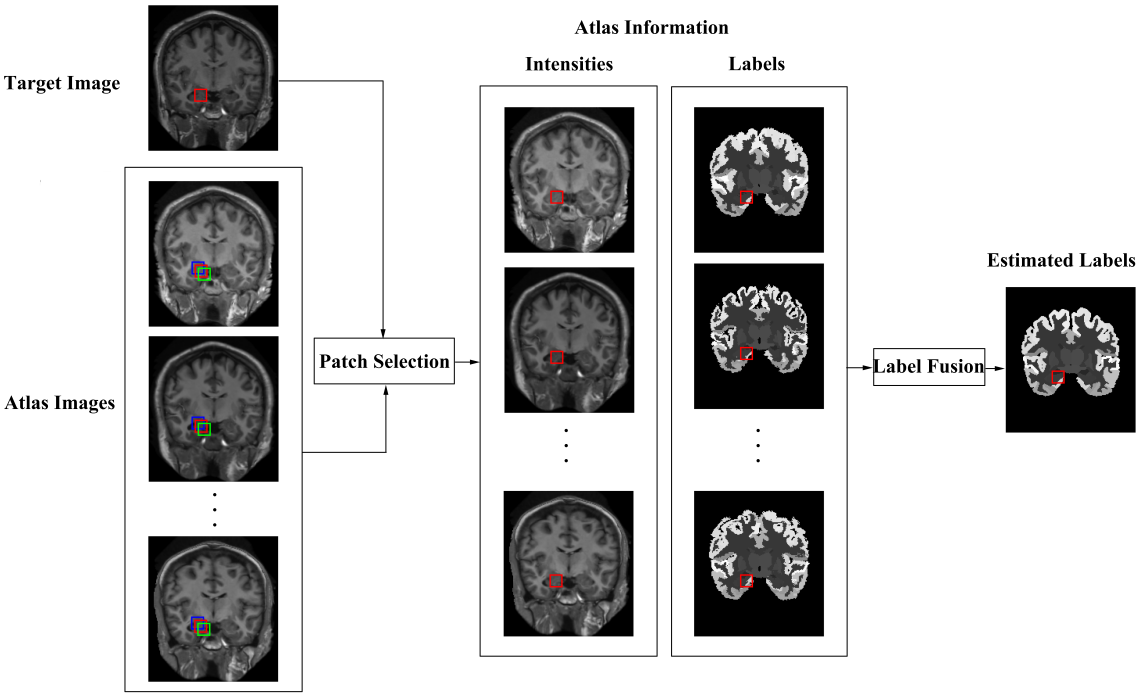


Figure 3.1: Illustration of labeling for the target patch, where red square in target image denotes the target patch; the blue, pink and green squares in atlas image indicate patches in a searching window; and the best matched patch in each atlas is shown as red squares.

the target image (I_i is also called warped atlas image); and S_i is the corresponding manual segmentation of this atlas image. After combining the warped atlas images, a fused label map is generated which can be considered as the segmentation of the target image.

Figure 3.1 illustrates the generation of labels for the target patch of the proposed method. First, the atlases (intensity and label image) are pairwisely registered to the target image. Then, for each atlas image, a patch selection scheme is performed to choose the patch in each atlas with the highest similarity with the target patch. Finally, by applying the label fusion algorithm to the patches with the corresponding location of the patches in atlas images, we obtain the estimated label of each patch. The approach is applied for every voxel in the target image to obtain the labels for the entire target image.

3.2.1 Patch Selection

The performance of atlas-based segmentation can be moderately improved by applying a local searching technique [25]. Although deformable registration has been performed before label fusion, the correspondence obtained from the registration may not guarantee the maximal similarity between the patch in the target image and that in the warped atlas image. Therefore, local searching within a small neighborhood around the voxel in the warped image is performed to achieve the maximal similarity.

Summed squared distance (SSD) is used to measure the similarity between the target patch and atlas patch. The SSD of the target patch centered at x and the atlas patch centered at x' is shown below.

$$SSD(x, y) = \|I_T(\mathcal{N}(x)) - I_i(\mathcal{N}(x'))\|^2 \tag{3.1}$$

where $x' \in \mathcal{N}'(x)$ with $\mathcal{N}'(x)$ a local searched neighborhood. Equation (3.1) indicates that given a patch $I_T(\mathcal{N}(x))$ in the target image and $I_i(\mathcal{N}(x))$ in the ith atlas image, it is possible to find a patch $I_i(\mathcal{N}(x'))$ whose center belongs to the neighborhood $\mathcal{N}'(x)$. The patch centered at x^i, which is called locally searched optimal correspondence, has higher similarity with the target patch than other patches with centers inside the neighborhood $\mathcal{N}'(x)$. Thus, the locally searched optimal correspondence is

$$x^i = argmin_{x' \in \mathcal{N}'(x)}[SSD(I_T(\mathcal{N}(x)), I_i(\mathcal{N}(x')))] \tag{3.2}$$

where $I_i(\mathcal{N}(x'))$ is the patch in the ith atlas image centered at x' with a radius r, and $I_T(\mathcal{N}(x))$ is the target patch centered at x with a radius r. x' is the voxel in the local neighborhood $\mathcal{N}'(x)$ with a radius r_s. By calculating the SSD between the patches in the target and the atlas images, we obtain x^i, which is the location from the ith atlas with the best image matching for the location x in the target image.

3.2.2 Label Fusion and Validation

Majority voting: After patch selection, n patches are selected as the candidates of voting for the target patch. The likelihood of that x taking label l can be computed

by counting the number of occurrence for l from $x_i, i \in 1, 2, \ldots, n$. Then, the label for x in the target image can be determined by choosing the label with the highest posterior probability. The final label $L(x)$ is obtained by

$$\hat{L}(x) = argmax_{l \in \{1,\ldots,\mathcal{L}\}} \sum_{i=1}^{n} p(l|A^i, x) \tag{3.3}$$

where x indexes through image voxels; $p(l|A^i, x)$ is the posterior probability that atlas A^i votes for the label l at x. Typically, deterministic atlases have unique label for every location, which means $p(l|A^i, x) = 1$ if $S_i(x) = l$, and 0, otherwise.

Improvement on majority voting: The label of the center voxel of the target patch can be produced using majority voting. However, since we have chosen the most similar patch to the target patch from each atlas image based on the intensity information, these selected patches can be considered to have similar segmentation to the target patch. For each voxel in the target patch, we can find a candidate voxel from the corresponding position in each selected patch, and thus, the label of each voxel in target patch can be determined by performing Equation (3.3) from its n candidate voxels. Given a three-dimensional image, for every patch with a radius r in the target image, $(2r + 1)^3$ voxels within the patch will be labeled by performing the above majority voting scheme. However, due to the overlapping among the target patches, each voxel in the target image have $(2r + 1)^3$ candidates after the majority voting. As a result, we apply another majority voting scheme to fuse the labels from the $(2r+1)^3$ candidates for the overlapped positions. Therefore, the modified majority voting scheme is a two-stage label fusion strategy where the estimated segmentation of each target patch is obtained at the first voting stage while the voxelwise prediction is obtained by fusing the candidate labels for the overlapped positions at the second voting stage.

Validation: The kappa index (Dice coefficient or similarity index) was computed by comparing the manual segmentations with those obtained with our method. For two binary segmentations A and B, the kappa index was computed as

$$\kappa(A, B) = \frac{2|A \cap B|}{|A| + |B|} \tag{3.4}$$

In quantitative MR analysis, manual segmentation is usually considered as a gold standard. The segmentation quality was estimated with the Dice coefficient by comparing the expert-based segmentations with the automatic segmentations.

3.3 Experimental Results

The proposed approach is applied to segment the hippocampus using T1-weighted MR images. The dataset used in the experiment includes 35 brain MR imaging scans obtained from the OASIS project. The manual brain segmentations of these images were produced by Neuromorphometrics, Inc., using the brain-COLOR labeling protocol. The dataset was applied in the MICCAI 2012 Multi-Atlas Labeling Challenge, where 15 subjects were used at the atlases and the remaining 20 images were used for testing.

In the experiment, we perform pairwise registered transformations between the atlas and the target images, as well as between each pair of the atlas images. The ANTs registration tool was used in this study to implement pairwise registration [10]. The antsApplyTransforms with linear interpolation was applied to generate the warped images, and the antsApplyTransforms with nearest neighbor interpolation was applied to generate the warped segmentations.

3.3.1 Impact of the Size of 3D Patch and Search Volume

The proposed method has two parameters, r for the local patch radius and r_s for the local searched neighborhood. The influence of these parameters are studied by evaluating a range of values $r \in \{1, 2, 3\}$; $r_s \in \{1, 2, 3, 4, 5\}$ in the experiment. First, we studied the impact of the patch radius on segmentation accuracy. The mean dice overlap coefficient results are shown in Figure3.2. Using the patch radius of $r = 1$, the algorithm performs much better than using larger patch radius. The segmentation accuracy also improves with the increase of the searched radius r_s. However, the

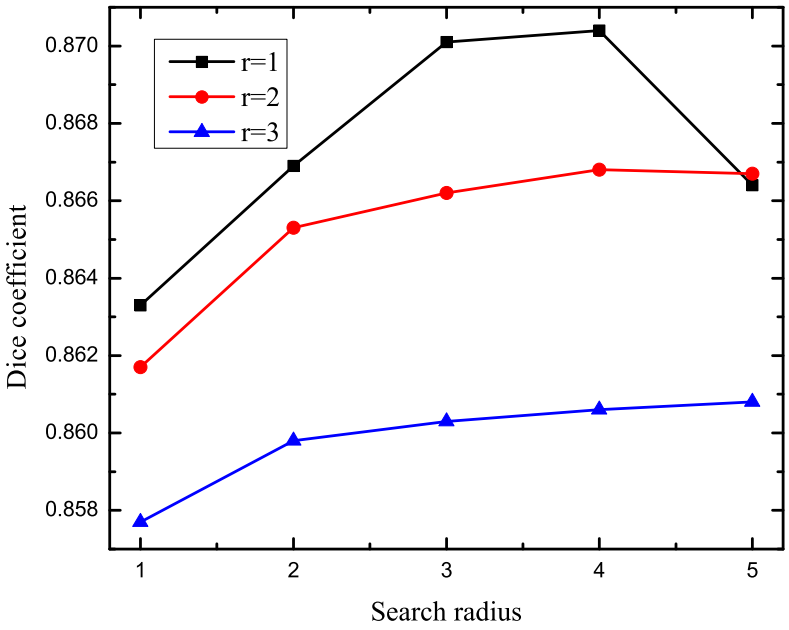


Figure 3.2: Hippocampus segmentation performance using different patch radius and searched patch radius.

Dice coefficient decreases when the searched radius $r_s > 4$. Larger searched radius improves the probability to find a similar patch with target patch, however, it also leads to an increase of mismatches. Figure3.2 indicates that the best Dice coefficient is obtained at $r = 1$ and $r_s = 4$. Figure3.3 shows the segmentation results for different sizes of local patch and searched patch.

3.3.2 Comparison Results in Hippocampus Segmentation

The average Dice overlap between automatic segmentation and manual segmentation for testing data is measure in the experiment. We compared our results with three automatic approaches, i.e. majority voting, global weighted fusion, and STAPLE [91]. The dice overlap coefficient of the left and the right hippocampus by the proposed approach is 0.8473 ± 0.0325 and 0.8447 ± 0.0370, respectively, and the average overlap is 0.846 ± 0.03. The box plot is shown in Figure3.4, where the central mark is the

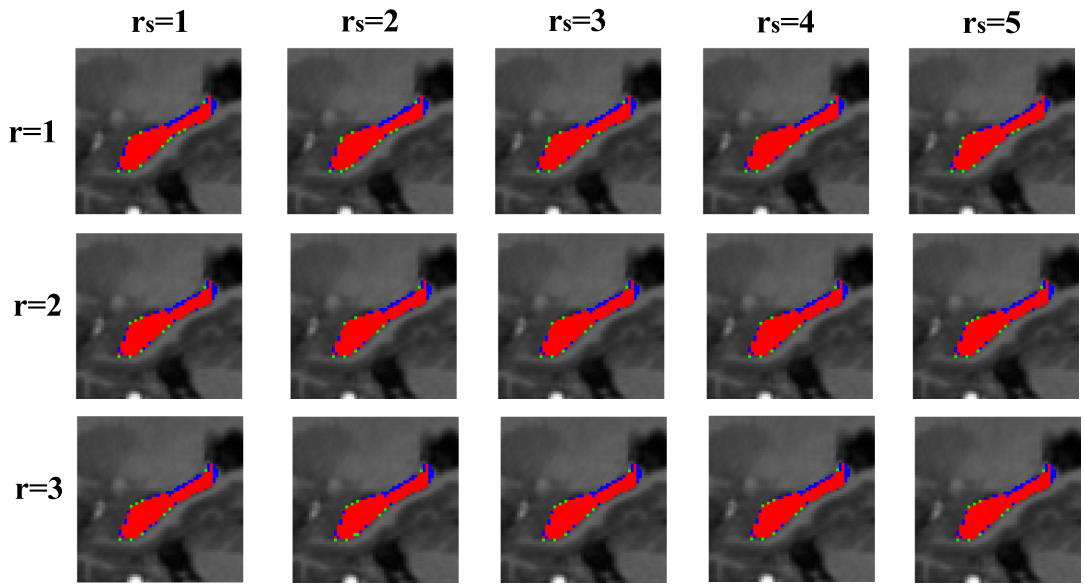


Figure 3.3: Sagittal views of the segmentations produced by different patch radius and searched patch radius. Where the red region shows the overlap between the automatic and the manual segmentation; the green region is the manual segmentation; and the blue region is automatic segmentation using the proposed method.

median, the edges of the box are the 25th and 75th percentiles. The whiskers extend to 2.7 standard deviations around the mean, and the outliers are marked individually as a '+'. As a comparison, the average Dice overlap obtained by majority voting, global weighted fusion, and STAPLE are 0.821, 0.807, and 0.836, respectively [91]. It is clear that the propose technique yield more than 1.2% Dice overlap improvement. In addition, the results of other three approaches were obtained by conducting the experiments in a leave-one-out strategy on a data set containing 39 subjects, while our approach use only 15 subjects as atlas set. Overall, the proposed method performs better in segmentation accuracy while using significantly fewer atlases than the reported methods.

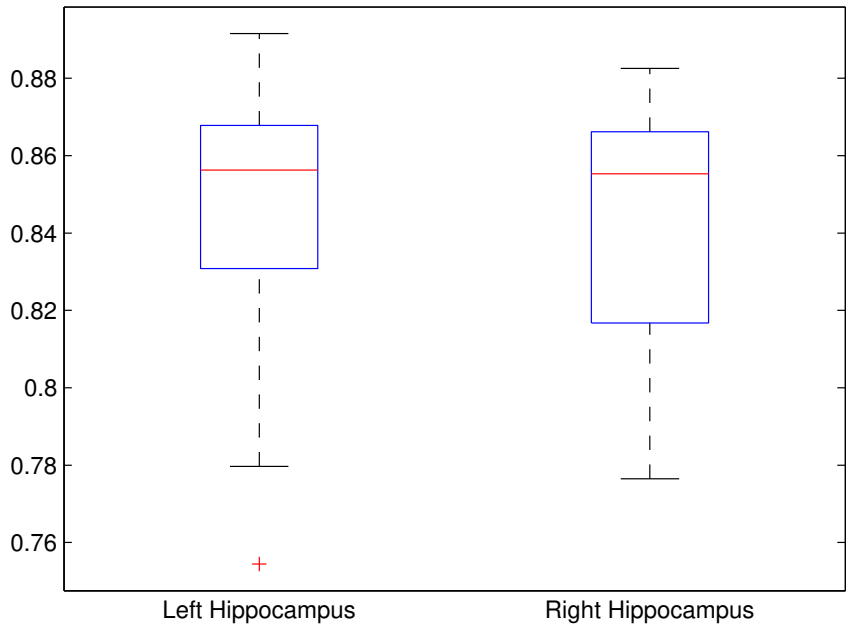


Figure 3.4: The dice overlap coefficient of the left and right hippocampus

3.4 Discussion and Conclusion

A novel approach to automatically segment anatomical structures based on the majority voting method is proposed. A patch selection strategy is proposed to ensure that the patch in the atlas with the highest similarity to the target patch is selected as the voting candidate. The proposed approach is verified by experimental evaluations on a standard dataset. Compared with three benchmark techniques, the segmentation results are significantly improved by the proposed method.

Chapter 4

Supervoxel Based Method for Multi-Atlas Segmentation of Brain MR Images

Although multi-atlas segmentation has been widely applied to the analysis of brain MR images, the state-of-the-art techniques in multi-atlas segmentation are strongly dependent on the pairwise registration. In this chapter, a new segmentation framework based on supervoxels is proposed to solve the existing challenges of previous methods. The supervoxel is an aggregation of voxels with similar attributes, which can be used to replace the voxel grid. By formulating the segmentation as a tissue labeling problem associated with a MAP inference in MRF, the problem is solved via a graphical model with supervoxels being considered as the nodes. In addition, a dense labeling scheme is developed to refine the supervoxel labeling results, and the spatial consistency is incorporated in the proposed method. The proposed approach is robust to the pairwise registration errors and of high computational efficiency. Extensive experimental evaluations on three publically available brain MR datasets demonstrate the effectiveness and superior performance of the proposed approach.

4.1 Introduction

An alternative method for segmentation is to formulate it as the energy minimization associated with the graphical model, in which the atlases are used to provide the prior knowledge about the spatial constraints [107]. As a result, the graphical model is usually considered as a post-processing technique to refine the results of the multi-atlas label fusion [3, 64, 84].

However, as an important pre-processing step of multi-atlas segmentation, the pairwise registration plays a crucial role in learning the spatial prior information for the MAP inference methods. Both the label fusion method and energy minimization method heavily rely on the complicated pairwise registration. Although the patch-based technique is effective in accounting for the registration errors, the performance is affected by the search radius of the local neighborhood in the registered atlases [26]. Increasing the search radius will greatly increase the computational cost while using a small search radius is not effective enough to remedy the registration errors.

To address these limitations, a graphical model-based multi-atlas segmentation algorithm from the supervoxel perspective is proposed. Superpixel-based MRF framework has been successfully applied to the semantic natural scene segmentation [104]. Inspired by [104], we extend it to the brain MR image segmentation using a supervoxel graphical model. The supervoxel is an aggregation of voxels with similar attributes, and thus, we can assume that the voxels within the supervoxel have the same label. Based on this assumption, each node in the graphical model is associated with a supervoxel, and the label minimizing the energy function is thus assigned to each element voxel within the supervoxel.

We propose a supervoxel graphical model for the whole brain segmentation in this chapter, and to ensure that the supervoxel segmentation fits the tissue boundaries, we propose to apply the supervoxel segmentation on label images rather than on the intensity images or feature images that are usually used in other studies. In addition,

a post-processing step, based on a grid graphical model with a high order potential, is performed to refine the supervoxel labeling results. The major contributions of this work include:

1. The spatial consistency is encouraged in the proposed method. According to the definition of supervoxel, the labels within the supervoxel are spatially consistent. In addition, the label consistency between neighboring supervoxels is encouraged by the smoothness term in the energy function.

2. The proposed method is robust to the pairwise registration errors. It searches similar atlas supervoxels in the neighborhood and encodes the supervoxel similarity into the data term of the energy function. It differs from the patch-based technique in that the search radius is defined by the number of supervoxels instead of voxels, which results in a larger search range given a fixed search radius. Consequently, the spatial prior is acquired by the initialization of the data term, rather than the sophisticated pairwise registration, and the dependency on the complicated pairwise registration is greatly alleviated.

3. The proposed approach is computationally efficient. Since the supervoxels are used as nodes in the graph construction, the number of nodes decreases to around $1/n$ of that in the voxel-based graphical models, where n is the average size of the supervoxels. Moreover, thanks to the insensitivity to the pairwise registration, affine registration can be used as a substitute for deformable registration so as to reduce the pre-processing time.

The rest of this chapter is organized as follows. We derive the theoretical basis and describe the implementation details of our method in Section 4.2. The experimental evaluations on three datasets are presented in Section 4.3, where the influences of different parameter settings are studied, and the advantages of our approach in segmentation accuracy and low dependency on pairwise registration strategies over the other state-of-the-art methods are demonstrated. Discussions about the results and

the advantages over the patch-based technique and learning-based methods are given in Section 4.4, and the paper is concluded in Section 4.5.

4.2 Method

Let I_T be the target image $I_T = \{I_T(x)|x \in \Omega\}$, where x denotes the voxel. The goal of multi-atlas segmentation is to estimate a label map L_T which assigns a label $l_x \in \{1, \ldots, \mathcal{L}\}$ to each voxel in the target image, given K atlases $A_1, \ldots, A_K$ with $A_k = (I_k, L_k)$ where I_k and L_k are the intensity image and the corresponding label image, respectively. This problem can be solved via MAP estimation [91]

$$\hat{L_T} = \arg\max_{L} p\left(I_T, L_T; \{I_k, L_k\}\right) \tag{4.1}$$

where $p\left(I_T, L_T; \{I_k, L_k\}\right)$ is the joint probability of I_T and L_T given the atlases. MRF optimization is often posed as the task of finding the label map L_T that optimizes the MAP problem. The problem corresponds to minimizing the following objective function, known as MRF energy, which is defined over an undirected graph including node set Ω and edge set $\mathcal{E}$

$$E(\hat{L_T}) = \sum_{x \in \Omega} \theta_x(l_x) + \sum_{x,y \in \mathcal{E}} \theta_{xy}(l_x, l_y) \tag{4.2}$$

where the node set is referred to as the voxels in the target image while the edge set consists of the undirected edges in the graph connecting pairwise nodes. The unary data term $\theta_x(l_x)$ encodes the probability of observing label l_x at voxel x while the smoothness term $\theta_{xy}(l_x, l_y)$ measures the cost of assigning l_x and l_y to two neighboring voxels which are connected by the corresponding edge.

In supervoxel graph, the supervoxels are considered as the nodes, the energy function is thus defined as:

$$E(\hat{L_T}) = \sum_{s \in \Omega_s} \theta_s(l_s) + \sum_{s,t \in \mathcal{E}_s} \theta_{st}(l_s, l_t) \tag{4.3}$$

where the node set Ω_s and edge set $\mathcal{E}_s$ denote the supervoxels and edges connecting pairwise supervoxels, respectively. The data term $\theta_s(l_s)$ measures the cost of assigning label l_s to supervoxel s, and the smoothness term $\theta_{st}(l_s, l_t)$ penalizes the label inconsistency between the adjacent supervoxels s and t. The supervoxel labels are evaluated through minimizing the energy function (Equation (4.3)). Under the assumption that the voxels within the supervoxel have the same label, the supervoxel labels are propagated to the corresponding element voxels, and the estimation of the label map L_T of the target is obtained.

Figure 4.1 illustrates the framework of the proposed method. Before implementing the proposed method, pairwise registrations are performed between the target and each atlas. Then, in order to construct the supervoxel graph, supervoxel segmentations are applied on the target and atlases, respectively (Section 4.2.1). The details of the supervoxel graph construction are described in Section 4.2.2. Last, a dense labeling step is proposed to acquire the refined label map of the target in Section 4.2.3.

4.2.1 Supervoxel Segmentation

Supervoxel segmentation is the first step in the supervoxel graph construction. We use the use the simple linear iterative clustering (SLIC) [1] algorithm [1], an adaptation of k-means clustering method, to obtain the supervoxel segmentation in this work.

In the supervoxel graph, it is supposed that all the voxels within the supervoxel have the same label. As a result, we expect the supervoxel segmentation to cluster the voxels of the same structure. In the SLIC algorithm, the distance D, a weighted 5D Euclidean distance in $labxy$ space (the CIELAB color space [l a b] and spatial space [x y]), is used to measure the distance between the voxels and possible supervoxel center, then at each iteration, the voxels are clustered to the nearest supervoxel. This measure relies on the intensity similarity and spatial proximity between the voxel and

[1] `http://ivrl.epfl.ch/research/superpixels`

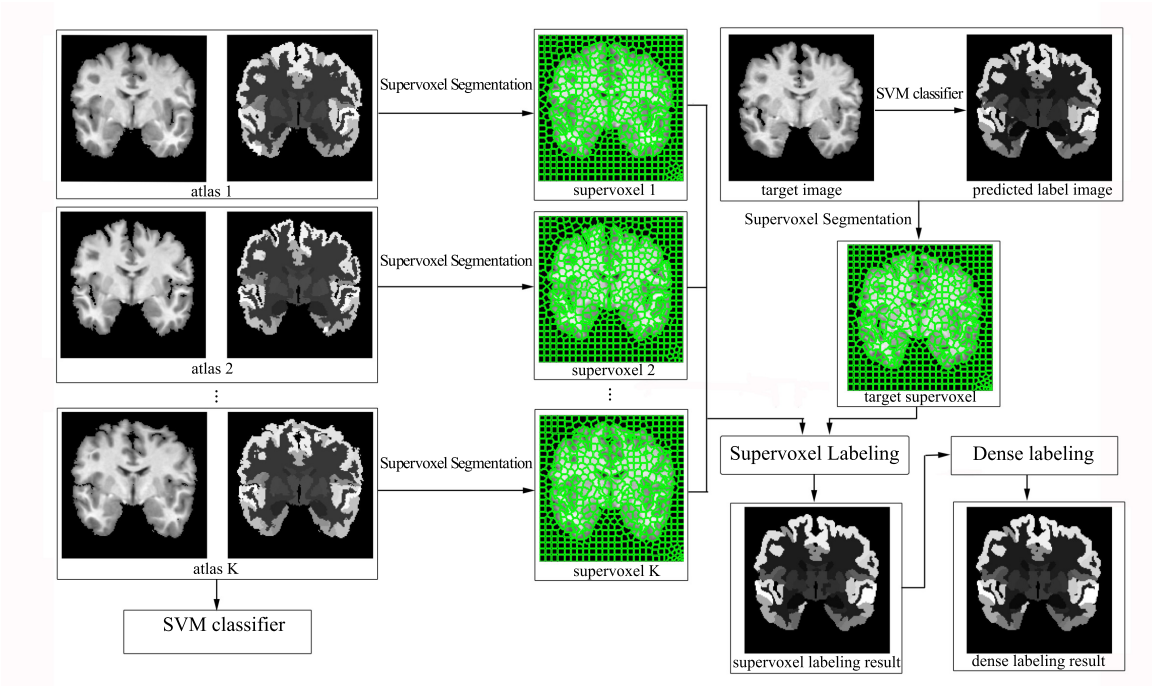
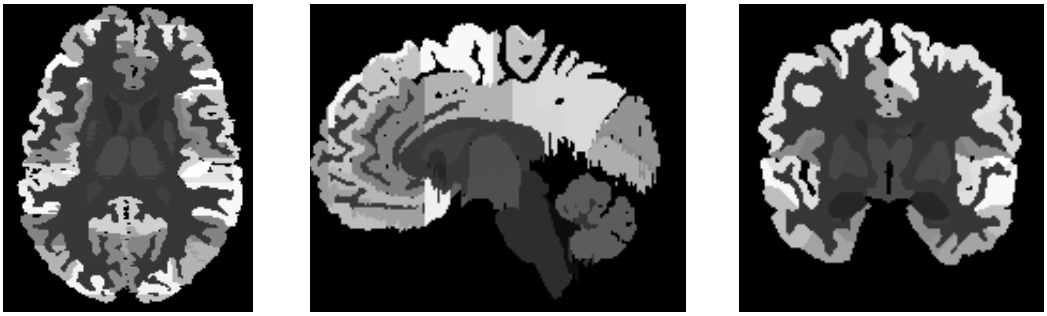


Figure 4.1: Framework overview of the proposed method. Supervoxel segmentation is performed on the target and the registered atlas images, respectively. The supervoxel labeling corresponds to a supervoxel-based graphical model. The dense labeling relates to a grid graphical model, aiming at refining the supervoxel labeling results. The SVM classifier is used to generate the predicted label image of the target for supervoxel segmentation and the probability map for initialization of data term in dense labeling.

the supervoxel center. However, intensity overlap among different structures widely exists in the brain MR images, so that performing SLIC algorithm upon the intensity images will increase the possibility of clustering the voxels of different structures into the same supervoxel. In contrast, in the label images, the voxels of a structure are represented by the unique value so that there is no intensity overlap existing. Consequently, it tends to cluster the voxels of the same structure into the supervoxel. Therefore, for the atlases, we apply the SLIC algorithm to the ground truth images. On the other hand, since the ground truth of the target is unknown, SLIC is performed on a predicted label image obtained via an support vector machine (SVM)[2] classifier for the target. To train the SVM classifier, voxel feature, which is a concatenation

[2] `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`



(a) Axial View (b) Sagittal View (c) Coronal View

Figure 4.2: An example of slices in the axial plane, sagittal plane, and coronal plane of the label image. Non-smooth tissue boundaries are displayed in the axial and sagittal plane.

of the texture, intensity, and position feature (see Table 4.1), is extracted from the unwarped atlas intensity images. By applying the SVM classifier, we obtain not only the predicted labels for aggregating voxels into supervoxels, but also the probability map to be used for calculating the data cost in the dense labeling stage.

In addition, the ground truth is manually annotated in the coronal plane for the atlases, which leads to non-smooth boundaries in both the axial and sagittal plane in the label images (Figure 4.2). The clustering of voxels in SLIC is based on the intensity similarity and spatial proximity. Involving spatial proximity encourages the voxels to be clustered to the spatially nearby supervoxel so as to form compact and nearly uniform distributed supervoxel segmentation. However, the non-smooth boundaries increase the difficulty in fitting the supervoxel segmentation to the boundaries, resulting in the accuracy decrease of 3D supervoxel segmentation. As a result, we technically perform superpixel segmentation on the coronal plane instead of supervoxel segmentation in the 3D space. To avoid misunderstanding, we still use the term "supervoxel" in this chapter.

Moreover, to calculate the data term, each supervoxel in the atlases should have the unique label. Therefore, a refinement scheme is proposed following the supervoxel segmentation of the atlases. First, we search for the supervoxels with multiple labels. For the clique with label l in the supervoxel, we assign the clique a supervoxel index

if the voxels in the clique are connective and the proportion of the clique size to the supervoxel size $p(l) > 0.5$. For each voxel without being assigned a supervoxel index, we merge the voxel v to a supervoxel s with label l which takes up the majority among the 8-neighbor of v. It should note that the label of v may not agree with that of s. As a result, we need to update the ground truth of v on top of merging v to s. Apart from enforcing the label consistency within the supervoxels, the refinement scheme also corrects the errors brought by the pairwise registration. As shown in Figure 4.3, some isolated holes are generated due to the interpolation procedure during the pairwise registration. By performing the refinement scheme, those isolated holes are filled, and the smoother boundaries of the tissues are obtained.

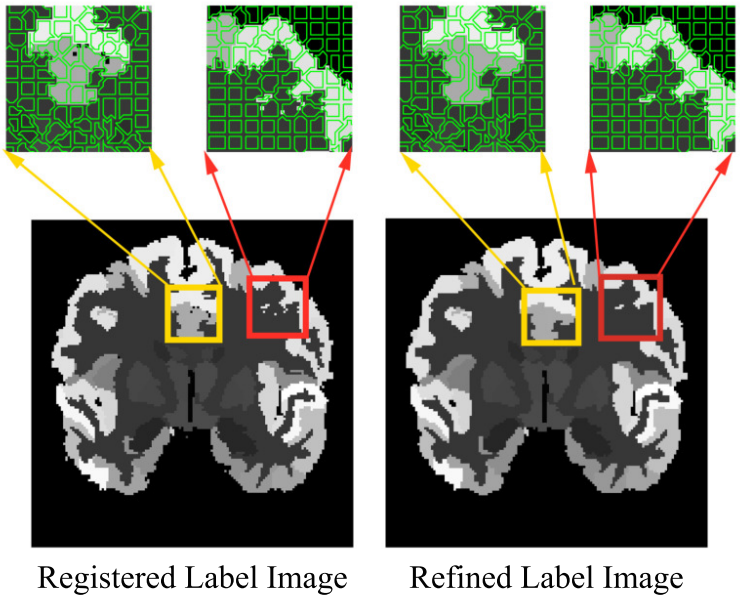


Figure 4.3: A comparison of the label image before (left) and after (after) the refinement scheme. Before performing the refinement scheme, the registered label image demonstrates isolated holes which cause label inconsistency within the supervoxel. After applying the refinement scheme, the isolated holes in the label image are filled, and the label consistency is enforced within the supervoxel.

In the implementation of the supervoxel segmentation, voxel feature vectors randomly selected from the unregistered atlas intensity images are used as the training samples to train the SVM classifier while the voxel feature vectors extracted from each voxel in the target intensity image are treated as the testing samples. In order

to avoid the feature in greater numeric ranges dominating those in smaller numeric ranges, the feature vector is scaled to $[-1, +1]$ before classifier training. The texture feature is normalized by the $L2$ norm while the voxel coordinate and the intensity feature are scaled to $[0, +1]$, respectively. The same scaling technique is applied to the testing samples. By applying the SVM classifier to the testing samples, a predicted label image and a probability map of the target are generated simultaneously. Then the coronal slices are extracted from the ground truth of the atlases and the predicted label image of the target, respectively, and the SLIC is applied to those coronal slices to acquire the supervoxel segmentation. Finally, to enforce the label consistency within the supervoxel for the atlases, we perform the refinement scheme on the supervoxel segmentation of the atlases.

4.2.2 Supervoxel Labeling

The unary data term $\theta_s(l)$ in Equation (4.3) is defined as:

$$\theta_s(l) = \sum_{s \in \Omega_s} -w(s, l)L(s, l) \tag{4.4}$$

where $w(s, l)$ is the weight computed through the Mahalanobis distance between s and the reference samples with class l, and $L(s, l)$ denotes the log likelihood score [104] for each supervoxel s and each class l.

$$L(s, l) = -\log \frac{p(s|l)}{p(s|\acute{l})} \tag{4.5}$$

where $\acute{l}$ is the set of all classes excluding l. Let D denote the set of all the supervoxels in the atlases, and N_s be the set of N nearest neighbors of s, which is found by searching the neighborhood with a search radius r in atlases and sorting the candidate supervoxels based on the feature similarity. In our implementation, we use the Euclidean distance to measure the feature similarity, and the supervoxel feature is a concatenation of four types of features (see Table 4.1). The feature description will

be given in Section 4.2.4. Thus, the likelihood score is defined as:

$$L(s,l) = -\log \frac{n(l,N_s)/n(l,D)}{n(\acute{l},N_s)/n(\acute{l},D)} \tag{4.6}$$

where $n(l,S)$ and $n(\acute{l},S)$ are the number of supervoxels with and without label l in set S, respectively.

The likelihood score $L(s,l)$ and weight $w(s,l)$ in our model jointly serve as the label prior and the intensity likelihood, which can be interpreted as the cost of assigning a label l to the supervoxel s from two perspectives. Since the contextual feature is one of the most important properties for medical images (e.g., the relative position of each tissue is fixed), we encode the contextual information into the likelihood score by assigning a probability to each possible label and excluding the impossible labels. However, there are two cases that the likelihood score is not comprehensive for computing the data cost: 1) the size of tissue is too small, and 2) the supervoxel is on the boundary of two types of tissues. In the first case, the target supervoxel is likely to be surrounded by the supervoxels of other tissues. In the second case, the target supervoxel is in the high contrast region while its candidates with the same label are in the low contrast region, and it is not easy to obtain correct matching between the target in the high contrast region and the candidates in the low contrast region based on the feature similarity. To sum up, in both the cases, it tends to yield a low likelihood score for the true label. Unlike the likelihood score computed from a limited number of local candidates, the weight $w(s,l)$ measures the distance from s to the center of class l by involving the samples randomly selected from all of the supervoxels in the atlases. The reason we use the Mahalanobis distance is that it accounts for the variance of each variable and the covariance between variables, which makes it meaningful in measuring the distance of data with multivariate distribution.

The smoothness term θ_{st} in Equation (4.3) estimates the cost of discontinuity of the label assignment in the adjacent supervoxels s and t. The Potts model is used to penalize the inequality of the labels with a constant:

$$\theta_{st}(l_s, l_t) = \lambda\delta(l_s, l_t) \tag{4.7}$$

where λ is the constant and $\delta(l_s, l_t)$ is the Kronecker delta.

The supervoxel graph construction is shown in Figure 4.4a. Edges added to the graph consist of $\mathcal{E}_1$ which denotes the edge between the adjacent supervoxels in the coronal plane and $\mathcal{E}_2$ which connects two supervoxels from the adjacent slices. Let $\{x_{s_n}, y_{s_n}, z_{s_n}\}$ denote the coordinates of the element of the supervoxel s. An edge $e \in \mathcal{E}_1$ connecting s and t is constructed if any element of t is 4-connected neighborhood of the element of s. Similarly, an edge $e \in \mathcal{E}_2$ is added between s and t if there exists t which includes an element t_n with $z_{t_n} \in \{z_{s_n} + 1, z_{s_n} - 1\}$. The number of edges associated with s is unfixed, which depends on the supervoxel connectivity in the inter-coronal plane and the intra-coronal plane. In this work, the MRF inference is implemented using sequential tree-reweighted message passing (TRW-S)[3] [67].

4.2.3 Dense Labeling

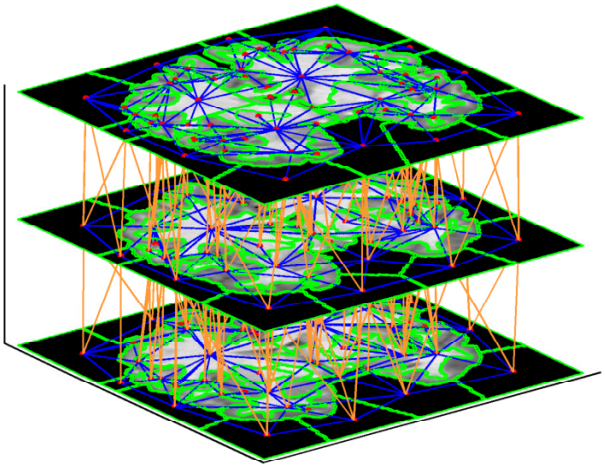
At the supervoxel labeling stage, we obtain a low resolution prediction of labeling. Because the supervoxel may contain multiple labels, some voxels, especially those whose ground truths differ from the others within the supervoxel, are likely to be incorrectly assigned a label during the supervoxel labeling stage. Therefore, in order to correct the mistaken labeled voxels, we perform a dense labeling strategy by introducing the high order potential defined in the robust P^N model [90].

The target image corresponds to a grid graph with 6 neighborhood system, and the objective function is designed as:

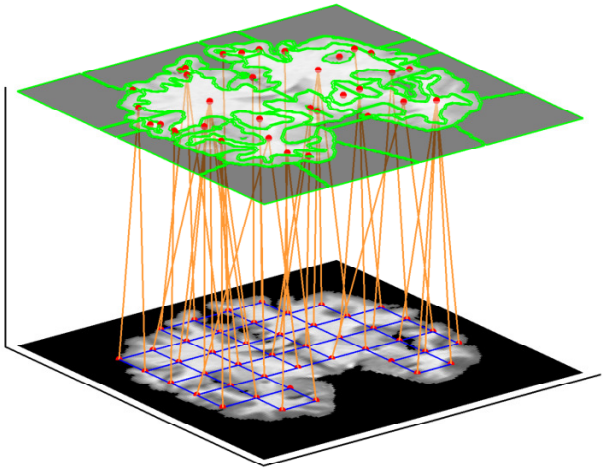
$$E = \sum_{x\in\Omega} \theta_x(l_x) + \sum_{x,y\in\mathcal{E}} \theta_{xy}(l_x, l_y) + \sum_{s\in\Omega_s} \theta_s^h(l_s) \tag{4.8}$$

where $\theta_x(l_x) = -\log p(x)$ is the data cost, with $p(x)$ the probability of the voxel

[3] `http://pub.ist.ac.at/~vnk/papers/TRW-S.html`



(a) Supervoxel Graph



(b) Dense Graph

Figure 4.4: Three consecutive slices are shown for the supervoxel graph (a), where the blue edges $\mathcal{E}_1$ indicate the pairwise potential in the coronal plane while the orange edges $\mathcal{E}_2$ are the pairwise potential of two adjacent slices. The dense graph (b) takes one slice as an example, where the bottom layer and top layer illustrate the grid graph and supervoxel layer, respectively. The blue edges indicate the pairwise potential in the grid graph while the orange edges show the high order potential. The nodes are indicated with red dots in both graphs.

x getting the label l_x, which is obtained from the same SVM classifier used in the supervoxel segmentation; $\theta_{xy}(l_x, l_y)$ is the smoothness term where the Potts model

is employed to penalize the label discontinuity of the neighboring voxels; the high order potential $\theta_s^h(l_s)$ takes the value $r_s^{l_s}$ if all the voxels within the supervoxel take the label l_s, otherwise, each inconsistent voxel is penalized with a constant. Unlike the robust P^N model [90], $r_s^{l_s}$ is initialized with $r_s^{l_s} = \min\{-\log(p_{l_s}), r_s^{max}\}$, where p_{l_s} is the one-hot representation of the supervoxel labeling result at s. Specifically, the vector $r_s^{l_s}$ equals to 0 at the dimension which corresponds to the supervoxel labeling result, and a constant r_s^{max} elsewhere. Based on the definition of $\theta_s^h(l_s)$, the high order potential is equivalent to the following formulation:

$$\theta_s^h(l_s) = r_s^{l_s} + \sum_{x \in s} \delta(l_x, l_s) = \theta_s(l_s) + \sum_{x,s \in \mathcal{E}} \theta_{xs}(l_x, l_s) \tag{4.9}$$

where $\theta_s^h(l_s)$ corresponds to a pairwise graph defined over the voxel x and the supervoxel s. The unary potential $\theta_s(l_s)$ relates to the cost of assigning l_s to s, and the pairwise potential $\theta_{xs}(l_x, l_s)$ corresponds to a Potts model that penalizes the voxels whose labels are inconsistent with the supervoxel label.

Therefore, the energy function of the dense labeling is defined over the variables $x \cup s$

$$E = \sum_{x \in \Omega} \theta_x(l_x) + \sum_{x,y \in \mathcal{E}} \theta_{xy}(l_x, l_y) + \sum_{s \in \Omega_s} \theta_s(l_s) + \sum_{x,s \in \mathcal{E}} \theta_{xs}(l_x, l_s) \tag{4.10}$$

The graph (Figure 4.4b) is composed of the grid nodes and supervoxel nodes. The edge set $\mathcal{E}$ consists of the edges which connect the 6 neighboring voxels in the grid and the edges which connect the supervoxel and its element voxels. To reduce the computational cost, the nodes in the graph are defined over the voxels and supervoxels which are not labeled as the background at the supervoxel labeling stage. Similarly, the optimization of the objective function is obtained via TRW-S [67].

Table 4.1: A complete list of features used in this work.

Type	**Feature**		**Dimension**
	Voxel	Supervoxel	
Texture	response of LMFB*	mean response of LMFB	38
Position	3D coordinate	3D coordinates of supervoxel center	3
Appearance	intensity	mean intensity of the supervoxel	1
	-	intensity histogram within the supervoxel	8

* LMFB: Leung-Malik filter banks [71].

4.2.4 Feature Extraction

Table 4.1 shows the voxel feature and supervoxel feature used in this study. The voxel descriptor used in the SVM classifier corresponds to a feature vector consisting of texture, position, and intensity feature. The supervoxel feature descriptor includes four components: texture, position of the supervoxel center, mean intensity, and the histogram of the voxel intensities within the supervoxel. The Euclidean distance of the supervoxel feature from the atlases to the target is computed so as to select the candidate supervoxels in the atlases and initialize the data term in the energy function. Since each type of feature shows different importance in differentiating tissues, we assign a weight w to each type of feature to improve the accuracy of the supervoxel matching.

The descriptor matching is considered as a convex optimization problem [100]:

$$\arg\min_{w \geq 0} \sum_{\substack{(x,y) \in P \\ (u,v) \in N}} \mathcal{L}(w^T \phi(x,y) - w^T \phi(u,v)) + \mu \|w\|_1 \tag{4.11}$$

where P and N are the matching pairs (i.e., x and y hold the same label) and non-matching pairs (i.e., u and v hold different labels); $\mathcal{L}(z) = \max\{z+1, 0\}$ is the hinge loss; $\|w\|_1$ is the regularization term; $\phi(x,y)$ denotes the sum of the squared difference of the each component of the descriptor. The elements of w are non-negative and a single weight w_i is applied to all the feature channels of each descriptor component. For instance, all the dimensions of the coordinates share the same weight. As the constraint is defined by the Euclidean distance in the descriptor space, which is same with feature similarity definition in Section 4.2.2, the weighted feature vector that

satisfies the constraint is more likely to improve the probability of matching supervoxel from the atlases to the target. We use the regularized dual averaging (RDA) method [125] to solve the regularized stochastic learning problem defined in Equation (4.11). More details about RDA can be found in [100, 125].

In our implementation, to construct the matching pairs, we randomly select equal supervoxel sample pairs per class from the registered atlases. Then the equal number of non-matching pairs are randomly selected. In this study, since the supervoxel feature is a concatenation of four components, a 4-dimensional vector $\phi(x, y)$, which is the sum of squared differences between the descriptors in the sample pair, is calculated so as to apply the RDA and learn the weight of each component of the feature.

4.3 Experiment

4.3.1 Evaluation

Two matrices are used in the evaluation: segmentation accuracy and under-segmentation error. The segmentation accuracy is evaluated by the Dice similarity coefficient between the ground truth and the segmentation result, which measures the overlapping ratio between two segmented regions and their average volume.

The under-segmentation error, which is a metric of the total number of "leak" caused by the supervoxels that overlap a given ground truth segment, is used for evaluating the accuracy of the supervoxel segmentation. Given a region from the ground truth segmentation g_i and the set of supervoxels required to cover it s_j, the under-segmentation error is expressed as:

$$E_{under} = \frac{\sum_{s_j | g_i \cap s_j \neq \emptyset} Area(s_j) - Area(g_i)}{Area(g_i)} \tag{4.12}$$

4.3.2 Pre-Processing

Before applying our approach, we performed the pre-processing steps in the following order for all the tests: bias field correction, pairwise registration, and image normalization.

First, the N4 bias field correction [106] was applied to the atlas and target images to correct the intensity inhomogeneity.

Next, the ANTs registration tool [10] was used to perform the pairwise registration between the target and atlas images. In this study, except for the Section 4.3.6 where a comparison of three pairwise registration strategies is conducted, all the experiments are performed on the deformable pairwise registered data. For the MICCAI 2012 dataset, we applied the data produced by the deformable registration in the ANTs, which was downloaded at http://placid.nlm.nih.gov/user/48. On the other hand, we performed the deformable pairwise registration on the LPBA40 and IBSR according to the steps mentioned at the beginning of this paragraph, but with a large convergence threshold and a small iteration number compared with the registration parameters used for the MICCAI 2012 dataset, aiming at reducing the pre-processing time.

Finally, decile normalization [94] was performed as follows: 1) The standard scale landmarks corresponding to each decile were calculated using the warped atlas intensity images; and 2) the image histogram (target and atlas intensity images) was mapped to the standard scale landmarks in a piece-wise linear fashion.

In order to reduce the computation time, we employ the skull-stripped data in this study. For the atlas image, the brain mask is derived from the ground truth image with label greater than 0. For the target image, the brain mask is generated from the predicted label image obtained from the SVM classifier.

4.3.3 Influence of Parameters

Parameters evaluated in this study include the SVM parameters, the supervoxel size, and the atlas number. We applied the empirical values on other parameters. The search radius r and candidate supervoxel number N are set to 10 and 50, respectively. The parameter of the Potts model λ, which is related to the smoothness term, is fixed at 10; and the threshold r_s^{max} associated with the high order potential is set to 30 for all the experiments.

Note that, the feature weight is learned from the matching pairs and non-matching pairs generated from the atlases. Consequently, the feature weight is learned based on each dataset, instead of being fixed for the whole study.

We applied the proposed method to the LPBA40 dataset to demonstrate the effect of different parameter settings. We randomly selected 20 subjects as the test data and the rest as the atlas subjects. Deformable pairwise registrations described in Section 4.3.2 were adopted before the parameter evaluations. The skull-stripped images provided by the dataset were used as the brain mask in this case. The feature weight vector learned by RDA is $\{0.681, 2.763, 1.482, 0.542\}$.

4.3.3.1 SVM parameters tuning

We use the radial basis function (RBF) kernel for the SVM, where c and γ are two key parameters. c is the penalty for misclassifying a data point while γ controls the 'spread' of the kernel and therefore the decision region. To estimate the parameters c and γ, we applied the grid search with value $c \in \{2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7\}$ and $\gamma \in \{2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2\}$ using five-fold cross-validation [19] on the training samples. Figure 4.5 shows that $c = 2^4$ and $\gamma = 2^{-1}$ achieve the highest classification accuracy of 85.75%.

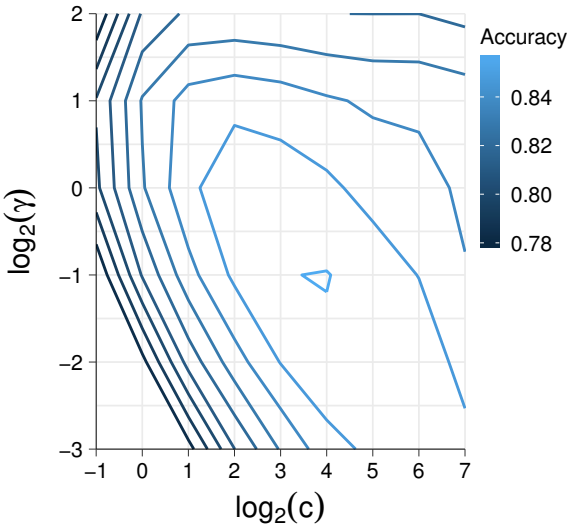
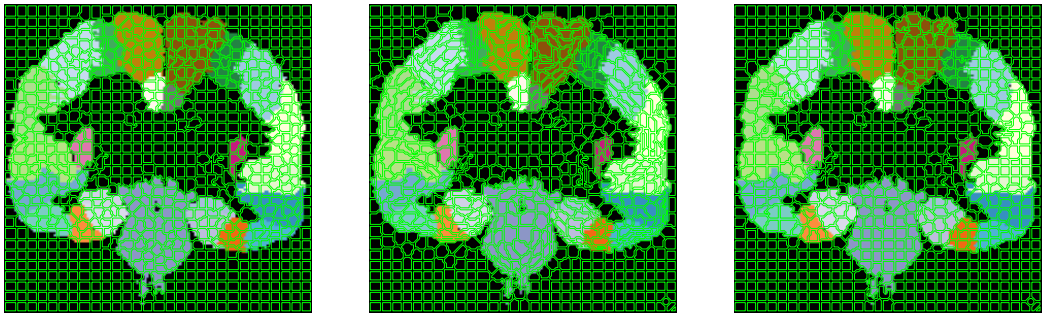


Figure 4.5: Influence of parameters γ and c in the SVM on classification accuracy.



(a) (b) (c)

Figure 4.6: Supervoxel segmentation results of using SLIC based on (a) intensity image, (b) feature image using a concatenation of the texture feature, coordinates and intensity, and (c) predicted label image obtained from the SVM classifier.

4.3.3.2 Influence of supervoxel size

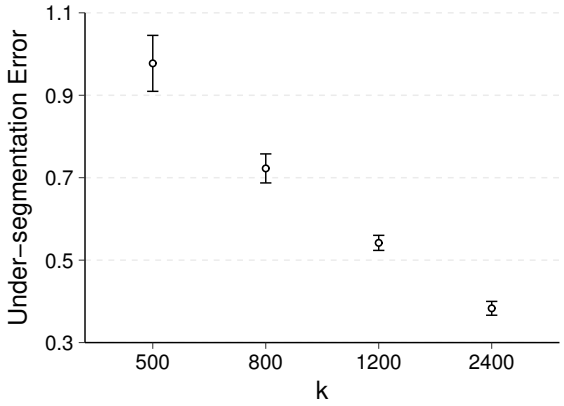
The accuracy of the supervoxel segmentation plays a critical role at the supervoxel labeling stage. We compared three types of images as the input of the SLIC algorithm: intensity image, feature image, and predicted label image. The feature image is defined in the voxel feature space (see Table 4.1) which consists of the texture feature, intensity, and the coordinates in the coronal plane. 50 coronal slices randomly selected from 20 test subjects were evaluated with the desired supervoxel number $k = 1200$.

Figure 4.6 shows the supervoxel segmentation results displayed on top of the ground truth images, where the label image outperforms the other two input images in terms of forming uniform supervoxels and fitting the boundaries of the ground truth. For quantitative analysis, the label image demonstrates the lowest under-segmentation error with 0.534 ± 0.001 in contrast to 1.049 ± 0.015 for the feature image and 0.630 ± 0.002 for the intensity image.

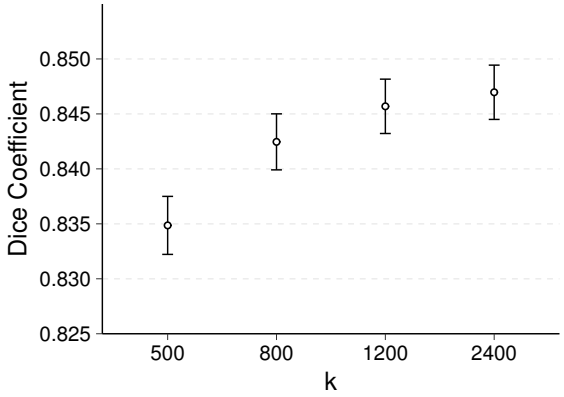
Another important parameter of our approach is the size of the supervoxel which is determined by k, the desired number of approximately equally sized supervoxels, in the SLIC algorithm. We performed our approach on the 20 test subjects with $k = \{500, 800, 1200, 2400\}$. Figure 4.7 shows the under-segmentation error, Dice coefficient, and the processing time, averaged over the 20 subjects, with respect to the value of k. Figure 4.8 demonstrates the supervoxel segmentation with different values of k on the ground truth image. It is evident that the small supervoxel size makes the over-segmentation to tightly fit the ground truth, and the segmentation accuracy increases with the increase of k as well as the processing time of the supervoxel segmentation. However, the computation time rises considerably when $k \geq 1200$. In this test, we make a tradeoff between the segmentation accuracy and the processing time, and set k to 1200 in the following experiments.

4.3.3.3 Influence of atlas number

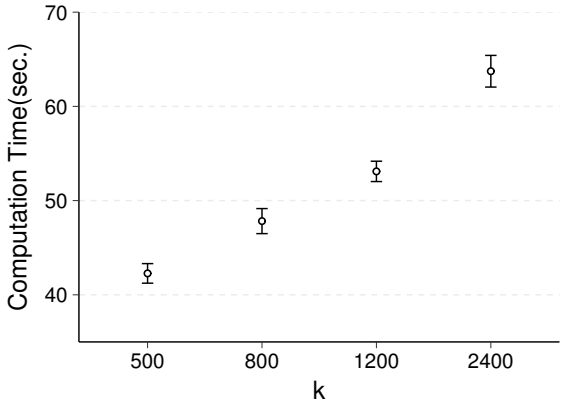
To study the influence of the atlas number, we performed the proposed method on the 20 test subjects with the number of atlases ranging from 1 to 20. Figure 4.9 demonstrates the averaged Dice coefficient with respect to the number of atlases. The segmentation accuracy improves with the increase of the number of atlas subjects while the increase rate reduces when the atlas number is greater than 10.



(a) Under-segmentation error

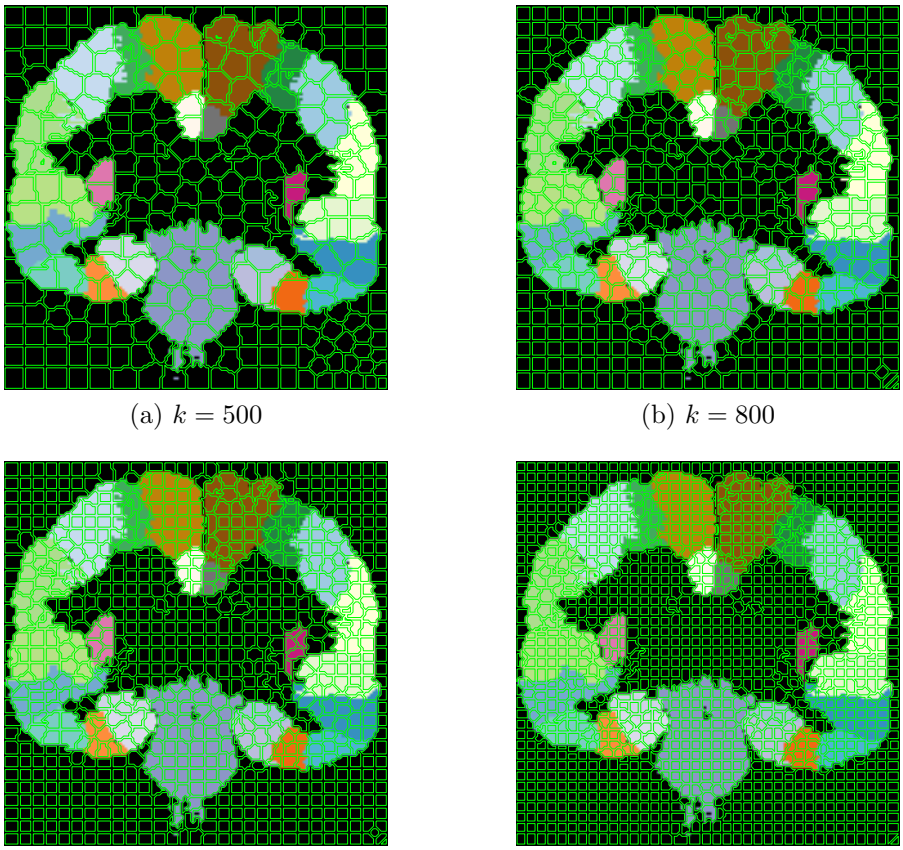


(b) Dice coefficient



(c) Processing time

Figure 4.7: Supervoxel segmentation performance with respect to k. (a) and (c) indicate the averaged accuracy and processing time for supervoxel segmentation (error bar at ± 1 std), respectively. (b) demonstrates the averaged segmentation accuracy for the dense labeling (error bar at ± 1 std).



(a) $k = 500$ (b) $k = 800$

(c) $k = 1200$ (d) $k = 2400$

Figure 4.8: Supervoxel segmentation on the ground truth image with different supervoxel size k.

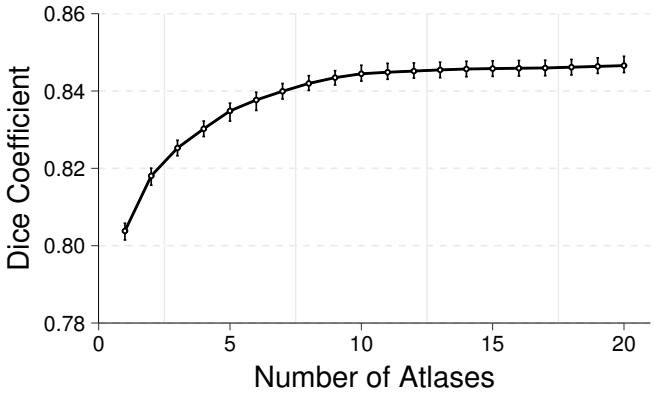


Figure 4.9: Overall accuracy in terms of mean Dice coefficient, with respect to the number of the atlases (error bar at ± 1 std).

4.3.4 Influence of Method Components

In this section, we investigate the influence of different components in the proposed method, and the results are summarized in Table 4.2 and Figure 4.10. The labeling results acquired from the SVM classifier are employed as the baseline. In order to study the contributions of the smoothness term and the effectiveness of the data term, we compare three results: 1) Labeling results by applying a graph model of 3D grids with 6 neighborhood system to the image, where the voxel probabilities obtained from the SVM classifier and same Potts model represent the data term and the smoothness term, respectively (SVM+MRF); 2) the labeling results that minimize the data term in Equation (4.4), which is equivalent to a label fusion scheme (supervoxel label fusion); and 3) the segmentation results of the supervoxel labeling in Section 4.2.2 (supervoxel MRF). For completeness, the refinement results in Section 4.2.3 are also included (dense MRF).

As shown in Table 4.2, there is a clear increase from the SVM (0.794 ± 0.028) to SVM+MRF (0.821 ± 0.063), and from the supervoxel label fusion (0.817 ± 0.009) to supervoxel MRF (0.839 ± 0.009). The results indicate that the graphical model is effective in improving the segmentation accuracy compared with that just fuses the candidate labels or utilizes the classifiers. In addition, by comparing the results of supervoxel MRF and SVM+MRF, it is obvious that the data term, which encodes the contextual information and the label prior, is effective in representing the likelihood of the supervoxel, and contributes to 2.9% increase. By applying the dense labeling refinement scheme, the Dice score continues to increase to 0.848 ± 0.010. For the qualitative analysis in Figure4.10b, the results of SVM classifier miss the spatial consistency and result in a noisy label image. By applying the MRF graphical model, the spatial inconsistency is alleviated, however, due to the influence of its data term (Figure 4.10b), all the voxels of the middle frontal gyrus are mislabeled as superior frontal gyrus in Figure 4.10c. Compared with the results of SVM and SVM+MRF, the mislabeling of the middle frontal gyrus area does not exist in the results of supervoxel

Table 4.2: Dice coefficients of different components analysis.

Method	Cortical labels	Sub-cortical labels	All labels
SVM	0.772 ± 0.031	0.935 ± 0.013	0.794 ± 0.028
SVM+MRF	0.805 ± 0.069	0.958 ± 0.018	0.821 ± 0.063
Supervoxel label fusion	0.795 ± 0.010	0.950 ± 0.012	0.817 ± 0.009
Supervoxel MRF	0.825 ± 0.010	0.956 ± 0.012	0.839 ± 0.009
Dense MRF	0.834 ± 0.013	0.958 ± 0.015	0.848 ± 0.010

(a) Ground truth (b) SVM (c) SVM+MRF

(d) Supervoxel label fusion (e) Supervoxel labeling (f) Dense Labeling

Figure 4.10: Segmentation results of the different components analysis.

label fusion and supervoxel MRF (Figure 4.10d and Figure 4.10e). Furthermore, by employing the refinement dense labeling, the labels of the voxels on the tissue boundaries are corrected (Figure 4.10f).

4.3.5 Experimental Results on Three Public Dataset

In this experiment, we compared the segmentation accuracy of the proposed method with four baseline methods: majority voting (MV) [42], patch-based method (PB)

[25], SVM segmentation with augmented features [4] [12], and joint label fusion[5] [118]. MV is simple but yields a competitive result. PB is efficient in alleviating the dependency on deformable image pairwise registration. The patch size is set to $7 \times 7 \times 7$ and the search volume is set to $5 \times 5 \times 5$. JLF represents the state-of-the-art weighted label fusion method. The parameters presented here is the same as suggested in [118] with $\alpha = 0.1$, $\beta = 2$, $r_p = 2$ and $r_s = 3$. SVMAF is a learning-based method, which achieves a good performance in the segmentation of the cardiac MR images. The parameters of the SVM classifier in the SVMAF are set to $\gamma = 2^{-8}$ and $c = 2^1$ resulting from a grid search with $\gamma \in \{2^{-9}, 2^{-8}, 2^{-7}, 2^{-6}\}$ and $c = \{2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3\}$. The control point spacing, patch size, and search window size are set to $3 \times 3 \times 1$, $7 \times 7 \times 1$ and $5 \times 5 \times 1$, respectively, according to [12]. We performed the four baseline methods and the proposed method on the three datasets: the MICCAI 2012 dataset, the LPBA40 dataset, and the IBSR dataset. As described in Section 4.3.2, the five multi-atlas segmentation methods were applied on the same pairwised registered data to ensure the fairness of comparison.

4.3.5.1 Experimental results on MICCAI 2012 dataset

The MICCAI 2012 dataset includes 20 test subjects and 15 atlas subjects, with 134 labels. We train the SVM classifier with $c = 2^3$ and $\gamma = 2^0$ resulted from a grid search with value $c \in \{2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7\}$ and $\gamma \in \{2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2\}$ using cross-validation, and the feature weight vector learned by RDA is $\{0.412, 2.689, 1.50, 0.374\}$.

Table 4.3 lists the mean and standard deviation of the Dice coefficient for the cortical structures, subcortical structures and all labels. The mean Dice coefficient of the cortical structures outperforms the other baseline methods while the performance in subcortical structures is lower than JLF and SVMAF. The reason is that the segmentation of subcortical area is dependent on the intensity information. As a result, the JLF algorithm, which computes the fusion weight based on intensity similarity, outperforms the proposed method in several subcortical structures. Our proposed

[4] `http://wp.doc.ic.ac.uk/wbai/software/` [5] `https://www.nitrc.org/projects/picsl_malf/`

Table 4.3: Dice coefficient and running time of four baseline methods and the proposed method on three public datasets.

Datasets	Method	Cortical	Sub-cortical	All	Time
LPBA40	MV	0.763 ± 0.012	0.856 ± 0.027	0.777 ± 0.072	1 min
	PB	0.764 ± 0.012	0.854 ± 0.027	0.774 ± 0.013	25 min
	SVMAF	0.624 ± 0.026	0.794 ± 0.024	0.648 ± 0.025	15 min
	JLF	0.801 ± 0.009	0.881 ± 0.013	0.813 ± 0.009	60 min
	Proposed Method	$\mathbf{0.834 \pm 0.013}$	$\mathbf{0.958 \pm 0.015}$	$\mathbf{0.848 \pm 0.010}$	30 min
MICCAI 2012	MV	0.702 ± 0.028	0.797 ± 0.034	0.729 ± 0.026	1 min
	PB	0.709 ± 0.027	0.801 ± 0.031	0.734 ± 0.026	30 min
	SVMAF	0.715 ± 0.028	0.818 ± 0.029	0.745 ± 0.026	20 min
	JLF	0.731 ± 0.029	$\mathbf{0.825 \pm 0.030}$	0.758 ± 0.026	180 min
	Proposed Method	$\mathbf{0.748 \pm 0.017}$	0.816 ± 0.021	$\mathbf{0.764 \pm 0.015}$	45 min
IBSR	MV	0.551 ± 0.022	0.772 ± 0.026	0.606 ± 0.023	1 min
	PB	0.561 ± 0.024	0.768 ± 0.026	0.613 ± 0.024	25 min
	SVMAF	0.564 ± 0.025	0.732 ± 0.039	0.606 ± 0.029	10 min
	JLF	0.644 ± 0.037	0.841 ± 0.012	0.693 ± 0.031	45 min
	Proposed Method	$\mathbf{0.706 \pm 0.022}$	$\mathbf{0.852 \pm 0.016}$	$\mathbf{0.743 \pm 0.021}$	20 min

method achieves a significant improvement ($p < 0.01$, paired t-test) in the mean Dice coefficient compared with MV, PB, SVMAF, and JLF. In Figure 4.11, we present the per-label accuracy. The cortical structures and the subcortical structures are shown separately (Figure 4.11a and Figure 4.11b), the proposed method outperforms others in the cortical area, while it does not show a high Dice coefficient as expected in the subcortical labels. Figure 4.12 illustrates the ground truth and segmentation results of each method. Unlike the other baseline methods, the result of the proposed method does not contain the undesired "holes" in the anatomical structures, which indicates the advantages of our approach in maintaining the spatial consistency.

4.3.5.2 Experimental results on LONI-LPBA40 dataset

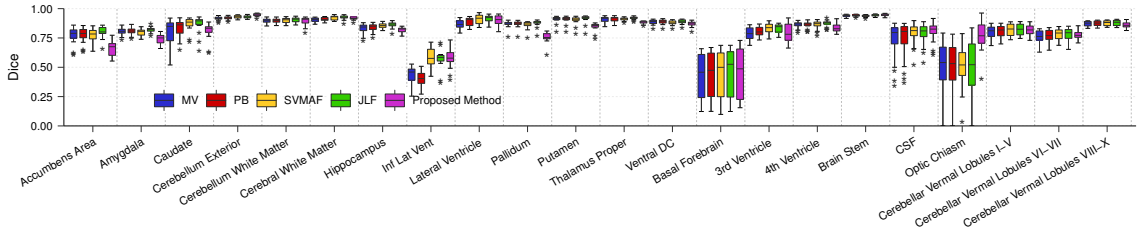
The parameters used for the LPBA40 dataset are consistent with those in Section 4.3.3. The Dice coefficient of each tissue is summarized in Figure 4.11, and the proposed method performs the best except for the precentral and gyrus rectus, and the proposed method achieves a significant improvement ($p < 0.01$, paired t-test) in the mean Dice coefficient of cortical structures, subcortical structures and the

overall labels compared with the baseline methods (Table 4.3). As with the MICCAI 2012 dataset, the high accuracy of segmentation in cortical labels is demonstrated in the LPBA40 dataset. In the six subcortical labels (we include brain stem and cerebellum as subcortical labels for simplicity), the proposed method also demonstrates significant improvement over the baseline methods. In Figure 4.12, MV, PB, and JLF show excessive smoothness of the boundaries while the proposed method does not. In addition, as we use a loose convergence parameter for the deformable registration, the segmentation quality of SVMAF yields a sharp decrease compared with its performance in the MICCAI 2012 dataset.

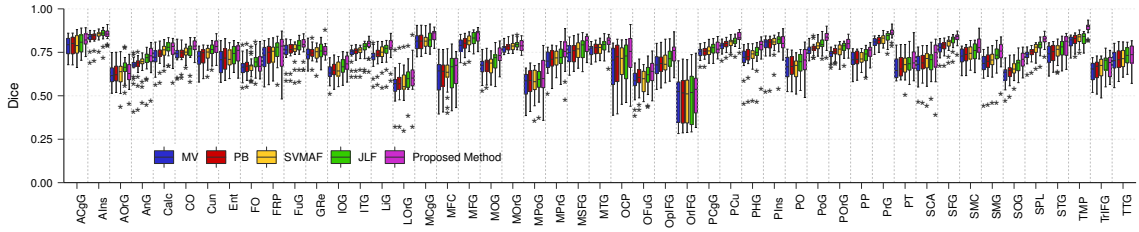
4.3.5.3 Experimental results on IBSR dataset

We randomly selected 9 subjects as the atlases and the remaining as the test subjects. The SVM classifier with $c = 2^2$ and $\gamma = 2^0$ is used to obtain the predicted label image by searching the grid of $c = \{2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7\}$ and $\gamma = \{2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2\}$, and the feature weight vector learned by RDA is $\{0.521, 2.234, 1.266, 0.548\}$.

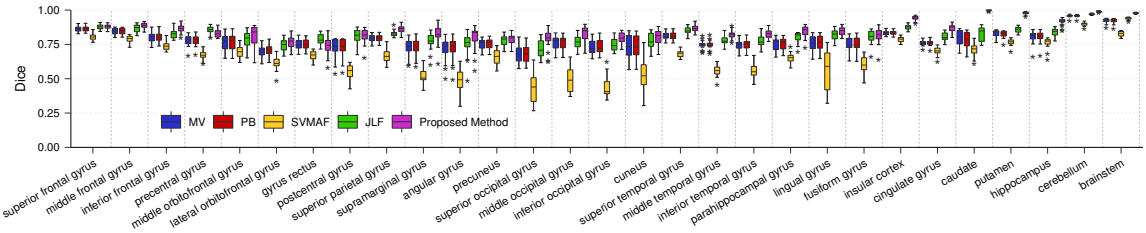
The proposed method performs the best in the mean Dice coefficient of cortical labels, subcortical labels and the overall labels, and it yields a significant improvement ($p < 0.01$, paired t-test) compared with the baseline methods, as shown in Table 4.3. However, our approach does not achieve the highest mean Dice coefficient in four subcortical structures: hippocampus, cerebral white matter, inferior lateral ventricle and CSF (see Figure 4.11). As with the results shown in the MICCAI 2012 and LPBA40 datasets, the performance in cortical structures is outstanding, which achieves the highest Dice coefficient in all of the cortical labels except T3p (inferior temporal gyrus, posterior). For the qualitative analysis in Figure 4.12, the baseline methods show excessive smoothness so that some tiny structures are missed in the subcortical area, and the spatial inconsistency exists in the cortical area. Like the results of the LPBA40 dataset, SVMAF does not yield a good performance due to the application of loose convergence parameter in the pairwise registration.



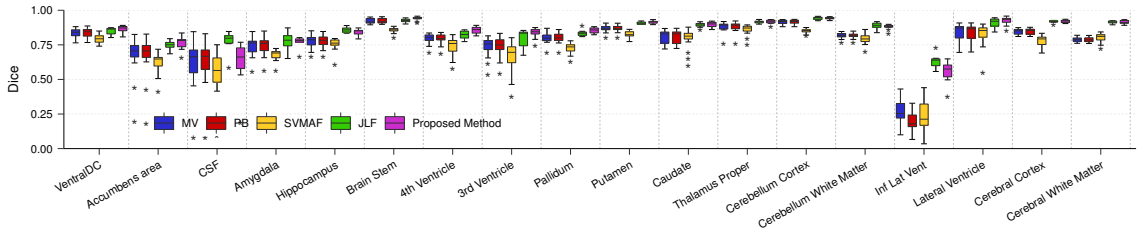
(a) Results of subcortical labels on the MICCAI 2012 Multi-Atlas Labelling Challenge



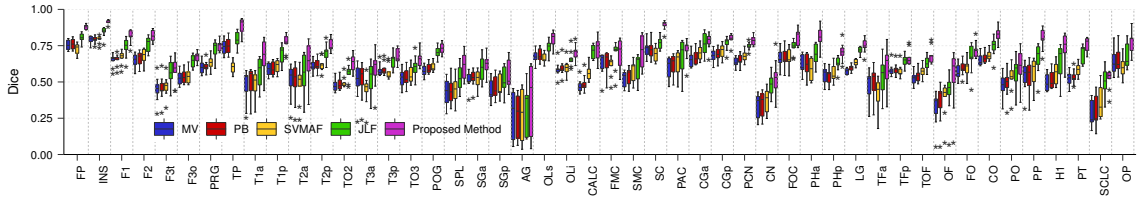
(b) Results of cortical labels on the MICCAI 2012 Multi-Atlas Labelling Challenge.



(c) Results on LONI-LPBA40 dataset



(d) Results of subcortical labels on IBSR dataset



(e) Results of cortical labels on IBSR dataset

Figure 4.11: Per-label accuracy comparison on the whole brain segmentation using three public datasets where the left and right hemisphere labels are shown jointly. The proposed method is compared with four baseline methods in the experiment.

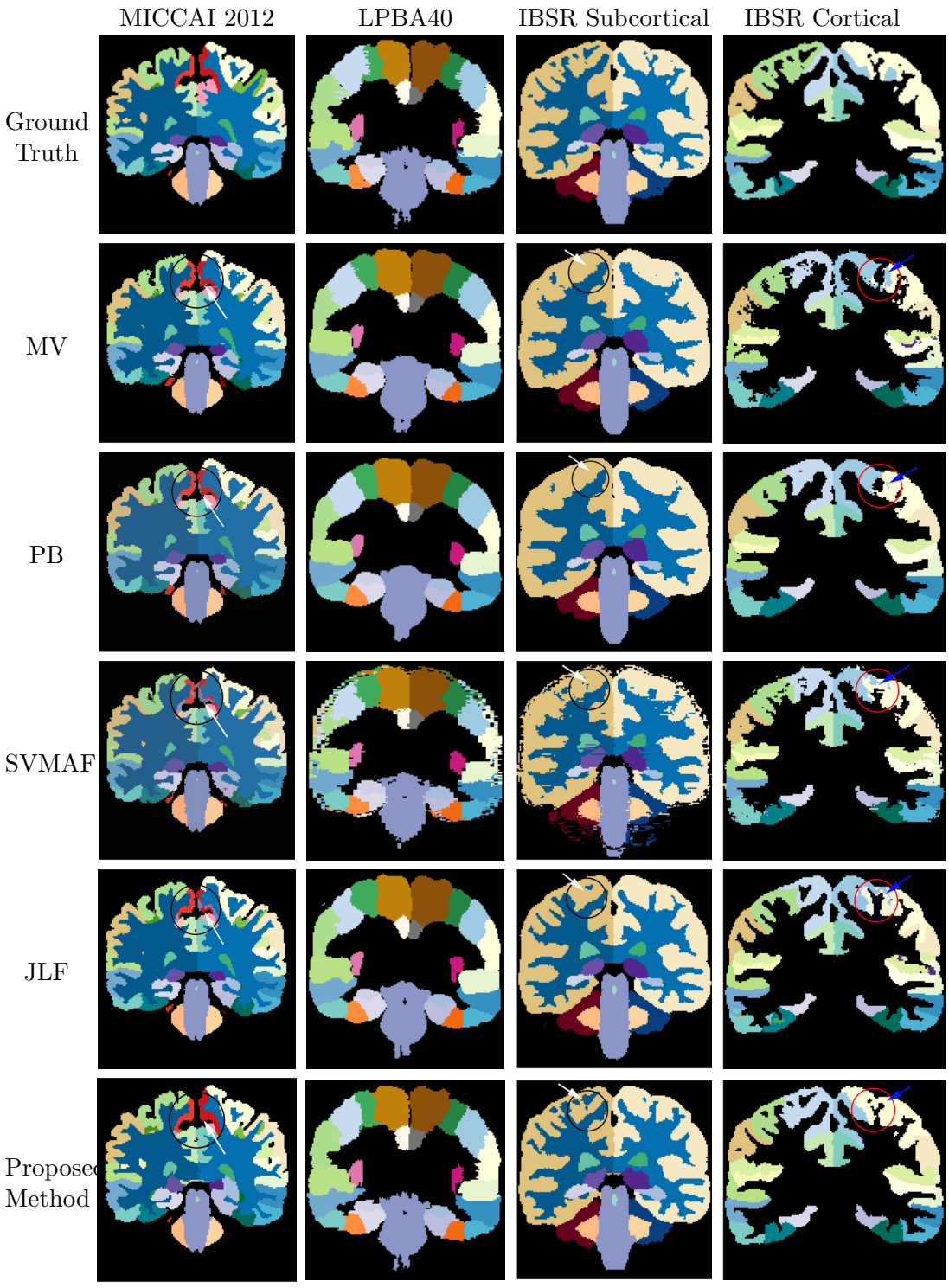


Figure 4.12: Segmentation results of the MICCAI 2012, the LPBA40, subcortical labels of IBSR, and cortical labels of IBSR datasets. Common mistakes (indicated by arrows) of the baseline methods include 1) spatial inconsistency in MICCAI 2012; 2) excessive smoothness of boundaries in LPBA40; 3) excessive smoothness in tiny structures in subcortical labels of IBSR; and 4) spatial inconsistency in cortical labels of IBSR.

4.3.6 Analysis of the Influence of Pairwise Registration Strategies

We analyzed the effect of applying different pairwise registration strategies on the LPBA40 dataset. We performed the tests with rigid, affine combined with rigid, and deformable registration strategies using the ANTs tool, with the cost of 1 min, 3 min, and 10 min for per pairwise registration. Then we applied the multi-atlas segmentation on the target image and corresponding warped atlas images. As shown in Table 4.4, the mean Dice coefficient is decreased by 0.8% (from 0.848 to 0.837) through shifting from deformable registration to the rigid registration for our method, while JLF and PB are decreased by 2.3% (from 0.813 to 0.794) and 11.8% (from 0.774 to 0.683). The proposed method does not show statistical significance ($p > 0.01$, paired t-test) in the pair of affine combined with rigid and deformable registration while JLF and PB do. Therefore, our method is less sensitive to the registration method. In other words, a less time-consuming registration method can achieve equivalent performance as those with complicated ones when incorporating our method.

Table 4.4: Dice coefficient of using different registration strategies

Method	Rigid	Rigid+Affine	Deformable
PB	0.683 ± 0.033	0.723 ± 0.016	0.774 ± 0.013
JLF	0.794 ± 0.014	0.796 ± 0.010	0.813 ± 0.009
Our method	0.837 ± 0.011	0.839 ± 0.009	0.848 ± 0.010

4.3.7 Computation Time

The average training time for the SVM classifier is 2 hours on a single desktop PC (i7-4900 CPU 3.60GHz, 16GB RAM). The testing time varies according to the number of atlases, class labels and the size of image. The running time per test is 45 min for the MICCAI 2012 dataset, 30 min for the LPBA40 dataset, and 20 min for the IBSR dataset, respectively. The baseline methods are evaluated using the same platform. Table 4.3 indicates that MV achieves the fastest running speed with only 1 min while

the JLF needs more than 1 hour per test. The running time for each subject of our proposed method is close to SVMAF and PB. Although the proposed method is slower than MV, it performs much better than MV in terms of the segmentation accuracy. The running time for the pre-processing steps is not considered in this section.

4.4 Discussion

The sensitivity of the proposed method is demonstrated with respect to the model parameters, the pairwise registration strategies, and the datasets. Generally speaking, the performance improves significantly by increasing the desired supervoxel number and the atlas number, however, the increase slows down after the number of k and atlases reach a certain value (i.e., 1200 for k and 10 for the atlas number on the LPBA40 dataset). For the pairwise registration, the insensitivity is verified by the experiments in Section 4.3.6. In the experiments of this work, the test subjects and the atlas subjects are acquired from the same scanner and labeled under the same protocol. However, further study is required to validate the performance of the proposed method when test data and atlases are acquired form different machines.

In addition, our approach achieves a significant improvement over the recently proposed state-of-the-art methods. The observed qualitative performance improvement demonstrates that the introduction of supervoxels contributes to preserving the spatial label consistency. As shown in the MICCAI 2012 dataset with 134 anatomical structures, the proposed method highlights its benefits in maintaining the spatial label consistency, particularly for the small structures. In addition, the proposed approach performs competitively in obtaining accurate details without excessively smoothing the boundaries, as demonstrated on the LPBA40 dataset. For the quantitative analysis, the performance in the cortical area is better than that in the subcortical area. Since the segmentation of the subcortical area is dependent on the intensity information, the JLF algorithm, which utilizes the intensity similarity to compute the weight, performs the best in several subcortical structures on the MICCAI 2012 and

IBSR datasets. For the LPBA40, a dataset with most labels within the cortex, the proposed method outperforms the baseline methods in almost all the labels.

Although both the proposed method and the patch-based method include the step of the local search, the aim of searching the neighborhood in the atlases is different. The proposed method computes the likelihood score through searching the neighborhood in the atlases so as to initialize the data term, while the local search is employed to compute the weight that the candidate label may contribute to the estimated label in the patch-based method. The data term is equivalent to a label fusion scheme which is employed by the other patch-based methods on a supervoxel level. By comparing the results of the supervoxel label fusion and other patch-based methods, the supervoxel label fusion achieves the highest segmentation accuracy 0.817 ± 0.009 compared with 0.774 ± 0.013 for PB and 0.813 ± 0.009 for JLF. In addition, apart from directly utilizing the intensity similarity, we also employ the texture similarity and spatial proximity in the feature descriptor, which alleviates the heavy dependency on intensity and contributes to preserving the label consistency.

The proposed method reduces the dependency on the pairwise registration in two ways. First, simple registration strategy (e.g., affine registration) could become a substitute for the complicated ones (e.g., deformable registration) since the registration errors are remedied by introducing the graphical model associated with the supervoxels. Second, the pairwise registration number per target decreases as it achieves competitive segmentation performance with a small atlas number. The learning-based methods do not require pairwise registration or adopt one registration per target, however, the training of the classifier requires a high computational cost in terms of time and storage. For example, the training time for CNN networks with two labels is approximately three hours on a workstation with an Nvidia Geforce GTX1080 GPU [92], and it needs three days for DeepNAT model with 25 brain structures [112]. Nevertheless, the spatial label inconsistency still exists in the segmentation results of the learning based methods. In addition, the accuracy of learning-based methods is affected by the number of training samples, and it needs to increase the number

of atlases or use data augmentation techniques. As mentioned in [112], the authors achieve comparable results with JLF by increasing the number of training scans from 15 to 20 in the MICCAI 2012 dataset.

4.5 Conclusion

In this work, we have developed a novel approach for the segmentation of the brain structure. The proposed method overcomes the challenges existing in the previous multi-atlas segmentation in terms of the computational efficiency and the dependency on the complicated deformable pairwise registration. The goals are accomplished by utilizing the graphical model associated with the supervoxels to solve the MAP estimation problem defined in multi-atlas segmentation. The proposed approach demonstrates superior performance over the state-of-the-art algorithms on three publically available datasets, and significant improvement was achieved in terms of overall accuracy, per-label accuracy, and qualitative assessment.

Chapter 5

AttentionNet: Brain Anatomical Structure Segmentation Using CNN with Attention Mechanism

Encoder-decoder network has been widely applied to medical image segmentation due to its performance in predicting the fine details. In this work, in order to further improve the ability to predict the fine details for the encoder-decoder like network, we propose an AttentionNet that exploits the attention mechanism to combine the shallow features from the down-sampling layer with the deep features from the up-sampling layer. The attention model finds the dependencies between the shallow features and the deep features, and selectively connects the important features to the up-sampling path. Furthermore, we develop an efficient decoder net that can be integrated with the state-of-the-art classification net. Extensive experiments demonstrate that the proposed AttentionNet significantly improve the segmentation performance compared with the other feature combination strategies on challenging brain segmentation datasets.

5.1 Introduction

Like the semantic segmentation, the goal of using ConvNets for medical image segmentation is to produce the pixelwise prediction. Inspired by the encoder-decoder

architecture that has been successfully applied to the semantic segmentation as well as the medical image segmentation [11, 87, 127], we design an efficient architecture for the anatomical structure segmentation of the brain MR images. The encoder is topologically identical to a classification network with the fully connected layers being replaced by the fully convolutional layers [23, 75], while the decoder is designed as a stack of up-sampling units [11, 87, 127].

The encoder outputs low-resolution feature maps because of the pooling or strided convolution operations. Directly up-sampling the coarse feature maps to the input resolution with a large stride results in a loss of the boundary details. To address this problem, the finer feature maps from the down-sampling path are connected to the up-sampling path in order to combine the high-level features with the local features. By combining the features that indicate the class with those carrying local information, the net is capable of making more precise predictions even for the pixels at the boundaries. Addition [127] and concatenation [80, 87] are the two most widely used strategies for feature combination, which have been demonstrated effectiveness in predicting the details of the segmentation. However, the existing feature combination methods did not consider the focused locations or enhanced the representations at the corresponding locations in the finer feature maps, resulting in limiting the performance of the feature combination.

Attention mechanism is a widely studied field in computer vision [56, 81, 115]. The nature of the selective visual attention is to direct the gaze towards salient objects in a cluttered visual scene [56]. Motivated by the idea of the attention mechanism in computer vision, we focus on developing a spatial attention model for feature combination in the encoder-decoder network, which enables the net to pay attention to the relevant spatial positions in the finer feature maps. The proposed spatial model is inspired by the scaled dot-product attention model [109], which was originally exploited in addressing the natural language processing problem. It connects the shallow features from the down-sampling layer and the high-level features from the up-sampling layer and captures the spatial dependencies between two feature maps.

Based on the spatial dependencies captured by the attention model, we can highlight those relative positions in the feature maps from the down-sampling path, softly select and connect those important features to the up-sampling path.

On the other hand, although 3D CNN is effective in capturing 3D contextual information, the 3D networks are usually designed in a shallow fashion because of the high requirement for the memory. Moreover, the training samples are cropped into small sub-volumes to enable the model to accommodate more mini-batches. In addition, there are rarely 3D CNN models pre-trained on a large dataset like the ImageNet, which makes it challenging to train a deep network on the small dataset. All of the factors mentioned above hamper the performance of the 3D CNN. To this end, we adopt the 2D CNN architecture and train the network on the 2D coronal slices. Moreover, to compensate for the loss of contextual information in the third dimension, we encode the position information into the inputs. The major contributions of this work include:

1. We develop and apply a spatial attention model to feature combination unit in encoder-decoder architecture. The attention model equips the net with the ability to highlight the important finer features and contributes to the precise dense predictions.

2. We develop a 2D CNN architecture, which benefits the model in terms of low memory requirement, deep architecture, and fine-tuning on the pre-trained model. The incorporation of the position information not only compensates for the loss of the spatial context in the third dimension but also enables the net to train on both intensity and spatial prior.

The chapter is organized as follows. We explain the design of the net architecture in Section 5.2. Then, we analyze the influence of the components in the AttentionNet in Section 5.3.3.3 and investigate the effect of the integration of the proposed attention model with the other modern classification networks in Section 5.3.4. Furthermore,

in Section 5.3.5, we compare the performance of the proposed AttentionNet with the state-of-the-art networks for medical image segmentation on three public datasets.

5.2 Methods

5.2.1 General Architecture

As illustrated in Figure 5.1, the net consists of three components: encoder, decoder, and feature combination. The encoder is topologically identical to a classification net which consists of a couple of down-sampling units. Each down-sampling unit stacks a few convolution layers and follows them with pooling layer (or strided convolution layer), resulting in down-sampling the spatial dimension of the input. The decoder is in charge of recovering the low-resolution, high-level feature maps to the full input size through a couple of up-sampling units. Each up-sampling unit includes an up-sampling operation (e.g., deconvolution layer or interpolation operation) to increase the spatial size. The feature combination unit associates the shallow, appearance features from the down-sampling unit with the deep, semantic features from the corresponding up-sampling unit. By combining the features that carry local information with those indicating the class, the feature combination unit outputs the features which not only are capable of localizing but also obey the global structure. Last, a fully convolutional layer is applied after the last up-sampling unit to obtain dense predictions.

5.2.2 Attention Model

The proposed spatial attention model finds the correspondence between the finer features and the high-level features via computing the dot-product. In this section, we start with introducing the original dot-product attention function, then derive the spatial attention function which is applicable to the encoder-decoder architecture for segmentation.

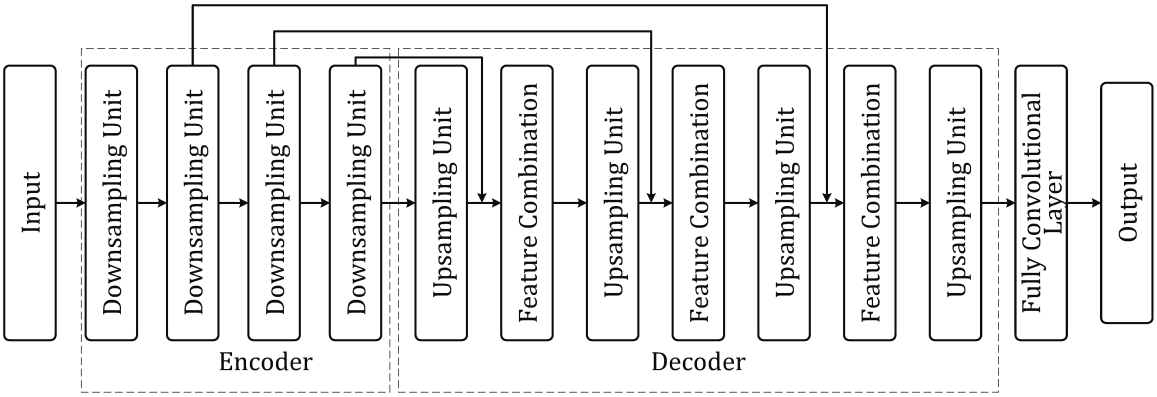


Figure 5.1: General encoder-decoder architecture for image segmentation.

5.2.2.1 Dot-product attention model

The attention model captures two information sources: query and query context, and a new representation is computed based on these two information sources. The query is associated with a query vector while the query context refers to a set of key-value pair, where the keys and values are all vectors. Generally, the dot-product attention model computes the similarities between the query vector and each key vector, then outputs a weighted sum of the values based on the similarity scores.

The dot product of two vectors has the geometric definition that represents the cosine of the angle between two vectors. Consequently, by performing the dot product of the query and all the keys we can obtain a bunch of similarity scores which indicate how each key aligns the query. Then the similarity scores are normalized by the softmax function, resulting in a weight map over all the values. As a result, the attention function outputs the weighted summation over all the corresponding values. The queries, keys, and values are packed together into matrices Q, K, and V, which enables the attention to be performed on a set of queries simultaneously. The dot product attention function [109] is:

$$Attention(Q, K, V) = softmax(QK^T)V \tag{5.1}$$

5.2.2.2 Spatial attention model

Based on the dot-product attention function, we propose a spatial attention function and apply it to the encoder-decoder network. For the spatial attention model, the feature maps of the up-sampling unit serve as the queries Q while the counterparts of the down-sampling unit serve as the keys K and the values V. The dot-product attention function computes a weight map that refers to the spatial correspondence between the two feature maps, aggregates the features over all the corresponding spatial positions in the finer feature maps, and combines the aggregated features with the up-sampling features. The structure of the building block for the spatial attention model is illustrated in Figure 5.2. The attention function is followed by batch normalization and ReLU activation. The shortcut connection is applied to the attention building block.

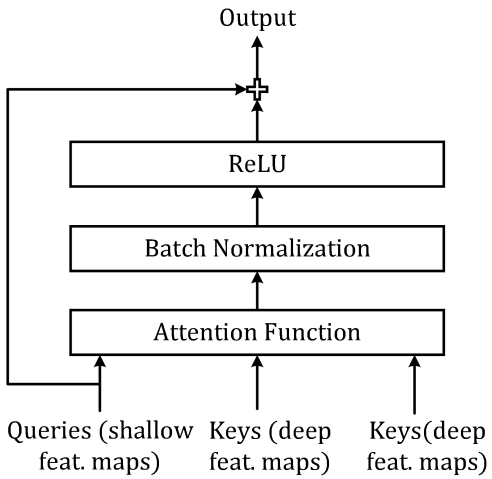


Figure 5.2: Building block of the spatial attention model.

Since the queries and keys are extracted from different feature spaces, we perform linear projections W_Q and W_K on the queries and the keys to transform them to the same subspace with dimension d_k. Moreover, to facilitate the "short connection" (see Figure 5.2), linear transformation W_V is applied to the values to match the dimension of the output of the attention function with that of the queries.

In addition, only when the two vectors are both unit vectors does the dot prod-

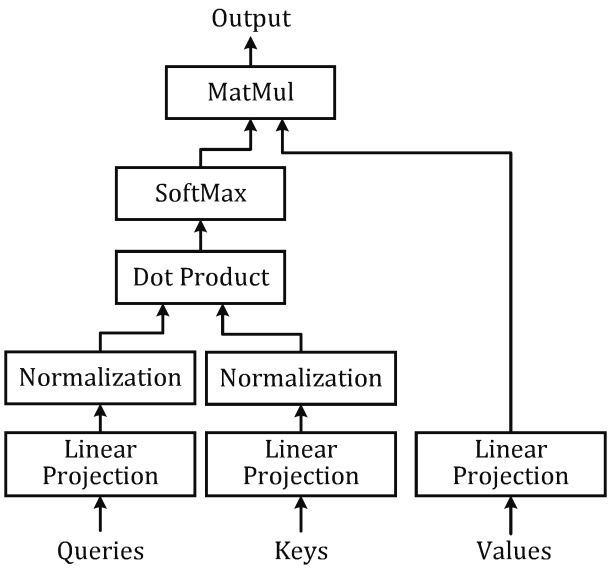


Figure 5.3: Building block of the spatial attention function.

uct measure the cosine similarity. We thus normalize the queries and keys before performing the dot product. The attention function is:

$$Attention(Q, K, V) = softmax(\sqrt{d_k}\frac{QW_Q}{\|QW_Q\|}\Big(\frac{KW_K}{\|KW_K\|}\Big)^T)VW_V \tag{5.2}$$

In Equation (5.2), we divide the queries and keys by their L2 norm to convert them to unit vectors. Another function of the softmax function is to enlarge the differences of similarity scores while the range of dot-product of the unit vectors is too small to take advantage of the softmax function. Therefore, we scale the dot product by $\sqrt{d_k}$ so that the results range from $-\sqrt{d_k}$ to $\sqrt{d_k}$. The spatial attention function is implemented as a building block of CNN, as shown in Figure 5.3. Equation (5.2) differs from the original dot product attention function [109] in terms of the normalization of the QW_Q and KW_K. We normalize the query and key vectors so that the dot product has the explicit meaning, which represents $\sqrt{d_k}$ times the cosine of the angle between the two vectors. The performances of Equation (5.2) and the original dot product attention is compared in Section 5.3.3.1.

However, it requires high memory consumption to force each query to attend all the positions in the key feature maps. Suppose the sizes of the queries and keys

are $N_q \times d_k$ and $N_k \times d_k$, respectively (where the 2D feature maps are flattened), performing a full attention function yields a $N_q \times N_k$ weight map. With the spatial resolution increasing after the up-sampling unit, the spatial sizes of the queries and keys both increase dramatically, resulting in a high memory requirement for the full attention.

Therefore, we develop a 2D attention model to make the query only attend the key positions within a block. To this end, we partition the queries (i.e., finer feature maps) and keys (i.e., high-level feature maps) into the same number of blocks and ensure the centers of the query block and the corresponding key block to be aligned. For each position in the query block, the spatial attention function Equation (5.2) is performed on the query and all the keys in the corresponding key block. Then by multiplying the values with corresponding spatial weight map, we acquire the weighted sum of the features over the positions inside the key block.

Figure 5.4 illustrates a toy example of the 2D attention module. Specifically, the queries (B, H_q, W_q, d_k) are partitioned into $\lceil \frac{H_q}{h_q} \rceil \times \lceil \frac{W_q}{w_q} \rceil$ ($\lceil \cdot \rceil$ is the ceiling function) blocks with block size of $N_q = h_q \times w_q$. Similarly, the keys (B, H_k, W_k, d_k) are partitioned into $\lceil \frac{H_q}{h_q} \rceil \times \lceil \frac{W_q}{w_q} \rceil$ key blocks with block size of $N_k = h_k \times w_k$, where $h_k = \lceil \frac{H_k}{H_q} \times h_q \rceil$ and $w_k = \lceil \frac{W_k}{W_q} \times w_q \rceil$. The 2D attention model computes a spatial weight map with $\lceil \frac{H_q}{h_q} \rceil \times \lceil \frac{W_q}{w_q} \rceil$ blocks, where each block is a $N_q \times N_k$ sub weight map that indicates the similarities between the queries in the query block and the keys in the corresponding key block. Because there exists a relationship between the size of the query block ($h_q \times w_q$) and the key block ($h_k \times w_k$), we only discuss the size of key block $N_k = h_k \times w_k$ in the rest of this chapter and investigate the effect of varying N_k in Section 5.3.3.2.

5.2.3 Architecture of the AttentionNet

By integrating the proposed spatial attention model with the encoder-decoder net, we obtain a new architecture "AttentionNet". Table 5.1 shows the architecture details of

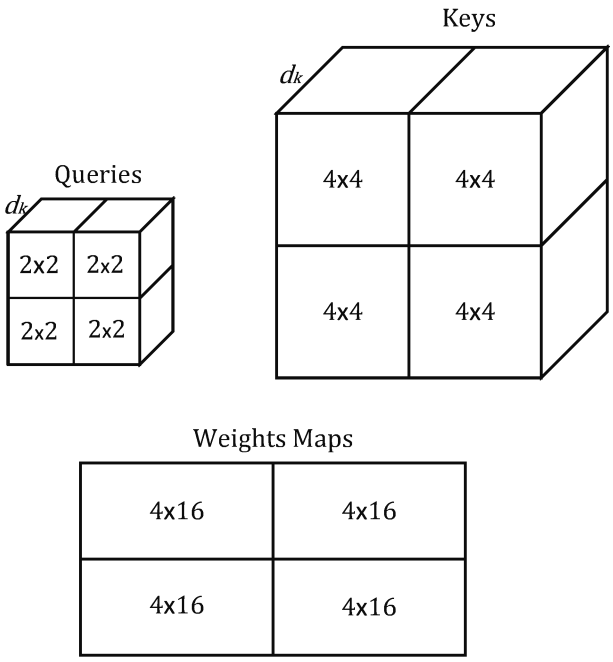


Figure 5.4: A toy example of the 2D attention model. For queries $(1, 4, 4, d_k)$ and keys $(1, 8, 8, d_k)$, they are partitioned into 2×2 query/key blocks, where the query block size is 2×2 and the key block size is 4×4. By performing the spatial attention function on the query block and the corresponding key block, we obtain a weight map with a size of 8×32. The weight map contains 2×2 sub weight maps, where each sub weight map with the size of 4×16 indicates the similarity scores of the query vectors and key vectors in the corresponding query and key block.

an AttentionNet that combines the ResNet-50 [41] with the spatial attention model (Attn-Resnet-50).

Residual block is the basic building block for the family of ResNet. Following the architecture of ResNet-50, we employ the bottleneck design for the residual block. The bottleneck block consists of three convolution layers, with kernel size of 1×1, 3×3, and 1×1. The two 1×1 layers are used to reduce (with the factor of 4) and then increase the dimension of the depth, and the 3×3 layer serves as a bottleneck with smaller input/output channels. Batch normalization is applied after each convolution and prior to the non-linear activation. The identity shortcut connection is applied if the input and output are of the same number of channels. Otherwise, the projection shortcut connection (1×1 convolution) is used to match the channels of the input with that of the output.

We remove the max-pooling layer that follows *conv1* in ResNet-50. In the down-sampling path, the down-sampling operation is performed four times by *conv1*, *conv3_1*, *conv4_1*, and *conv5_1* with a stride of 2. We hence stack four 2x up-sampling units to recover the feature maps to the input resolution. As shown in Figure 5.5, the proposed up-sampling unit, residual upsampling block (ResBlock), is designed as a deconvolution layer followed by a residual block. The deconvolution layer with 3×3 filter performs the 2x upsampling on the input and outputs 256-d feature maps. The same bottleneck design is employed in the residual block of the up-sampling unit.

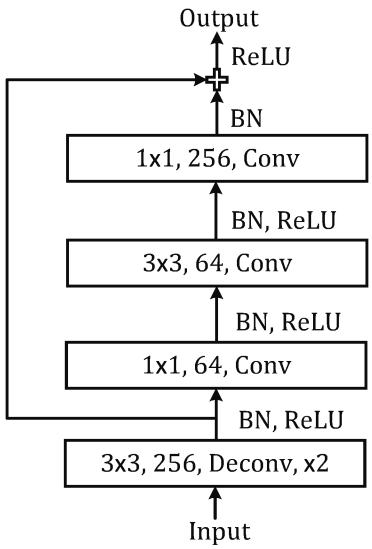


Figure 5.5: Structure of the residual upsampling block.

As shown in Table 5.1, three spatial attention building blocks are employed in this architecture, and each attention building block is inserted between two up-sampling units. For *attention1*, *attention2*, and *attention3*, the keys and values are the feature maps of *conv4_6*, *conv3_4*, and *conv2_3*, while the corresponding queries are the output feature maps of *upsample1*, *upsample2*, and *upsample3*. In order to decrease memory usage, all the up-sampling units produce feature maps with a depth of 256. To this end, the linear projection W_Q, W_K, and W_V transform the queries, keys, and the values to subspace with dimension 256 for all the attention layers. Last, a 1×1 convolutional layer is added on top of the last up-sampling unit to produce a pixelwise prediction.

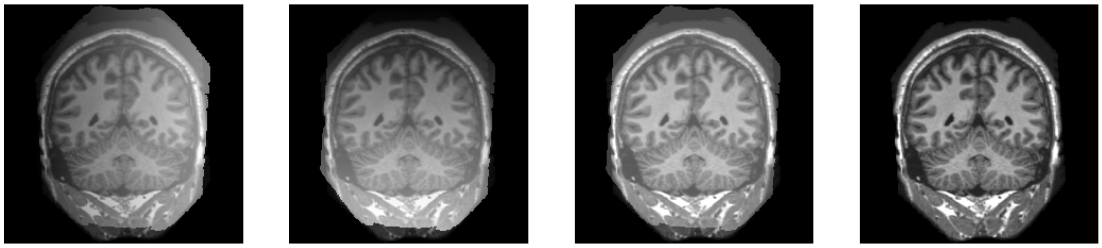
Table 5.1: Architecture of the Attn-Resnet-50.

Layer Name	Block Type	Output Size	Output Channel	Repetition No.
input	input	161×161	3	1
conv1	7×7 convolution	81×81	64	1
conv2_x	bottleneck	81×81	256	3
conv3_x	bottleneck	41×41	512	4
conv4_x	bottleneck	21×21	1024	6
conv5_x	bottleneck	11×11	2048	3
upsample1	res-upsampling block	21×21	256	1
attention1	attention block	21×21	256	1
upsample2	res-upsampling block	41×41	256	1
attention2	attention block	41×41	256	1
upsample3	res-upsampling block	81×81	256	1
attention3	attention block	81×81	256	1
upsample4	res-upsampling block	161×161	256	1
classification	1×1 convolution	161×161	No. of classes	1

For the AttentionNet, the repetition number of the down-sampling units differs from that of the corresponding up-sampling units. Moreover, the output channels of the up-sampling unit are not always equal to the corresponding down-sampling unit. As a result, the asymmetric architecture of the proposed AttentionNet has fewer parameters than the symmetric encoder-decoder architecture, which makes it easy to train on small datasets (e.g., medical images data).

5.2.4 Spatial Information

Previous studies show that the spatial prior provides valuable information for brain labeling [49] and demonstrate the importance of the spatial information for brain anatomical structure segmentation using CNN [28, 113]. In addition, since we employ the 2D CNN architecture, employing the spatial information can compensate for the loss of the contextual information in the third dimension. In this work, we augment the intensity image with the relative coordinates as the input.



(a) (b) (c) (d)

Figure 5.6: Coronal image augmented by the relative coordinates: (a) x, (b) y, (c) z. (d) is the original coronal image without position information.

To calculate the relative coordinates, we first compute the foreground mask of the brain volume, which is a non-zero mask of the 3D brain image. The relative coordinates are $(\frac{x-x_{min}}{x_{max}-x_{min}}, \frac{y-y_{min}}{y_{max}-y_{min}}, \frac{z-z_{min}}{z_{max}-z_{min}})$, where x_{min}, x_{max}, y_{min}, y_{max}, z_{min}, z_{max} are the minimum and maximum values of the foreground mask along x, y, and z directions. Then the relative coordinates are scaled to the range of $[0, 255]$. The relative coordinates provide the estimation of the position in the anatomical space for each voxel. We slice the three relative coordinate volumes along the coronal axis, and element-wise added the three sliced relative coordinate images to the three channels of the tiled coronal intensity image, resulting in an input image with three channels. Figure 5.6 illustrates an example of the input with position information embedded.

5.3 Experimental Results

5.3.1 Preprocessing

We apply the decile intensity normalization [95] to the volumetric MR images to deal with the intensity scale inhomogeneity with regard to intra- and inter-patient variations. Then, the 3D images are sliced into coronal images. After adding the relative coordinates, the input images are normalized to have zero mean by subtracting the mean value.

5.3.2 Implementation Details

We perform online data augmentation with random scaling, random rotation, and random cropping. The network is trained with Adam optimizer [63] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We use a mini-batch of 8 images and initial learning rate of 0.001. We apply an exponential decay to the learning rate (decay every 20 epochs with a base of 0.8). The model is trained for 150 epochs on the corresponding Imagenet-pretrained model. Except for the weights of the deconvolutional layers, all the weights are initialized as in [40]. For the deconvolutional layers, we follow the scheme in [75] to initialize the up-sampling to bilinear interpolation. Dice coefficient between the ground truth and the dense prediction produced by the network is adopted as the evaluation metric. All the networks are evaluated with 5-fold cross-validation on the corresponding datasets. Our CNN is implemented using the Tensorflow library on an NVIDIA GTX 1080 Ti GPU. Dice coefficient between the ground truth and the dense predictions is used as the evaluation metric.

5.3.3 Analysis of the Network Architecture

In this section, we quantitatively analyze the effect of the components of the proposed AttentionNet. All the models examined in this section use ResNet50 as the encoder net and are evaluated on LPBA40 dataset using the training scheme in Section 5.3.2. The parameters of the encoder net are initialized with model pre-trained on Imagenet.

5.3.3.1 Effects of normalization of the queries and keys

To investigate the effectiveness of the proposed spatial attention model, we compare our model with the original scaled dot-product attention model in [109] by evaluating the mean Dice coefficients. For the attention model in [109], the keys and queries are not normalized into the unit vectors while a layer normalization is applied after the attention function. Consequently, following the expression $Output = layernorm(Q +$

Attention(Q, K, V)) in [109], we construct the building block by applying a "short connection" to the output of the attention function and the queries and following it with a layer normalization. Note that no $L2$ normalization is employed in this original attention function.

Two attention models are integrated with the same encoder-decoder architecture (i.e., the down-sampling and up-sampling paths in Attn-ResNet50) and employ the 2D attention modes ($N_k = 4^2$ for all the attention layers). We train the two nets on LPBA40 dataset using the same training scheme as discussed in Section 5.3.2. The results show that the proposed attention model obtains the mean Dice coefficient of 0.853 ± 0.010, and yields 1.5% improvement over the attention model in [109], of which the mean Dice coefficient is 0.838 ± 0.015. Compared with the unnormalized vectors, the dot-product of normalized vectors explicitly represents the cosine similarities, resulting in the improvement of segmentation performance.

5.3.3.2 Size of key block

In this section, we investigate the effect of using different sizes of the key block N_k. N_k determines the size of the region in the key feature maps that the query vector attends. In this study, the square key block is employed for all the attention layers. We compare two schemes for N_k: 1) Unique N_k where all the attention layers are set to the unique size and 2) Staircase N_k where an increasing size from *attention1* to *attention3* is employed. Table 5.2 shows the mean Dice coefficients of the Attn-Resnet-50 on LPBA40 dataset for the two schemes of a range of different values.

The baseline model is the case of the unique block size $N_k = 1^2$, where the query vector only attends the same position in the finer feature maps. Compared with the baseline model, the staircase N_k significantly improves the performance to 0.864. When the unique block size increases to $N_k = 2^2$ and 4^2, the mean Dice coefficients significantly improve to 0.856 and 0.853, respectively, compared with the baseline. However, the performance decreases to 0.848 with using a larger unique block size of

Table 5.2: Dice coefficients of LPBA40 for Attn-Resnet-50 at different settings of N_k. Unique block size $N_k = 1^2$ is the baseline. Compared with the baseline, the staircase N_k, unique $N_k = 2^2$, and unique $N_k = 4^2$ achieve significant improvement, according to two-sided, paired t-test ($^{**}p < 0.005$,$^{*}p < 0.001$).

	Unique N_k				Staircase N_k
	$N_k = 1^2$	$N_k = 2^2$	$N_k = 4^2$	$N_k = 8^2$	$N_k = 2^2, 4^2, 8^2$, for *attention1,2,3*
Dice	0.850 ± 0.0100	$0.856 \pm 0.011^*$	$0.853 \pm 0.010^{**}$	0.848 ± 0.010	$0.864 \pm 0.010^*$

$N_k = 8^2$. We observe that the staircase block size outperforms any unique block sizes in this experiment. Furthermore, for the three cases of the unique N_k, small block size achieves better performance than the large block size.

Figure 5.7 shows the weight maps of three attention layers with different settings of N_k. For each weight map, the block indicated by the black square is the sub weight map that indicates the similarities between the queries in the query block and the keys in the corresponding key block.

In each block, i.e., the sub weight map, the diagonal values demonstrate how much a query vector correlates to the same position in the key feature maps. For the staircase N_k, we observe that most of the large weights concentrate on the diagonal for each attention layer, which agrees with the intuition that the queries are supposed to strongly correlate to the same position inside the key block. In contrast, the weight maps of the unique N_k do not demonstrate strong diagonal response. By comparing the weight maps of the three cases of unique N_k ($N_k = 2^2$, $N_k = 4^2$, and $N_k = 8^2$), we found that the weight map of the small unique N_k shows much more intensive diagonal values than that of the large unique N_k for *attention1*. Because the key feature maps of *attention1* (i.e., *conv4_6)* is repeatedly sub-sampled feature maps, enlarging the key block size is bound to involve a lot of distant and irrelevant positions and to allocate weights to those positions. As a result, the attention model pays increasing attention to the neighboring positions other than the same position in the key feature maps, which undoubtedly hinders the feature representation ability of the net.

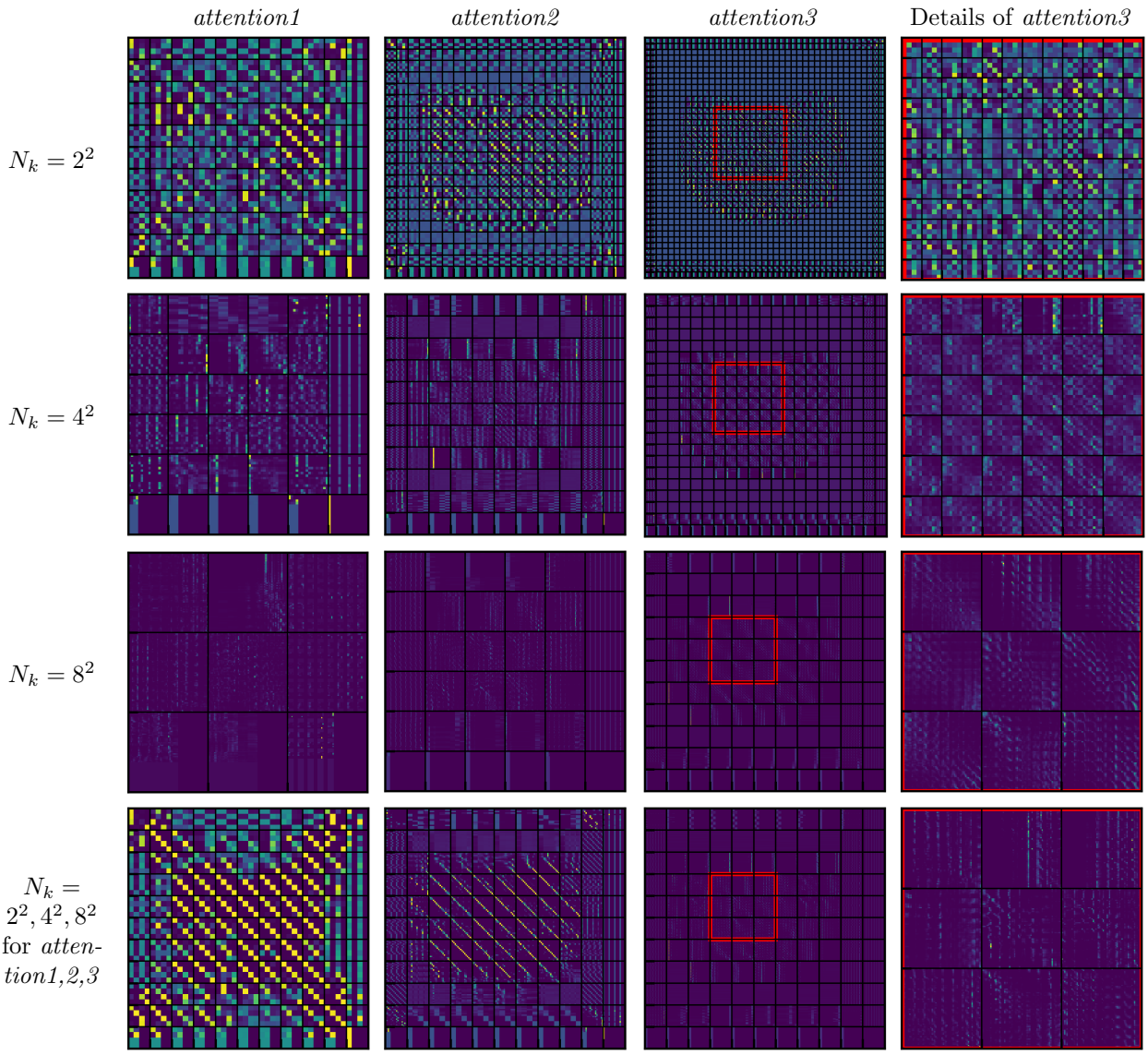


Figure 5.7: Visualization of the weights maps of three attention layers with different settings of N_k. Column 4 indicates the details of the red square in *attention3*.

However, for *attention3*, the weight values tend to be balanced for the small unique N_k while the diagonal values are dominant for the large unique N_k. It is because the feature difference is subtle among positions within the small neighborhood for the key feature maps of high-resolution (i.e., *conv2_3*). Consequently, using small unique N_k causes the attention layer to average the features over the positions in the block other than to select the correlated ones. We also infer that the balanced distributed weights in the last attention layer (e.g., *attention3* of $N_k = 2^2$) cause some

negative effects during the backpropagation, resulting in a cluttering weight map for the corresponding previous attention layer (e.g., *attention1* of $N_k = 2^2$). In contrast, as shown in Figure 5.7, the staircase N_k ensures that the dominant weight locates at the same position or its small neighborhood for each attention layer. The clear correlation between the queries and keys also reflects the ability of representative learning of the network.

5.3.3.3 Effectiveness of the AttentionNet

In this section, we quantitatively analyze the effectiveness of the AttentionNet. Resnet-50 is adopted as the encoder for all the nets. Moreover, all the nets are evaluated on LPBA40 dataset with 5-fold cross-validation using the training scheme discussed in Section 5.3.2.

First, to investigate the effectiveness of the spatial attention model, we compare the proposed spatial attention model with the other two feature combination variants: addition and concatenation. Addition [127] and concatenation [87] have been widely used in the existing encoder-decoder architectures and have shown the effectiveness in localizing the details in both semantic and medical image segmentation problems.

Then, we examine the performance of the residual upsampling block (ResBlock). Another two up-sampling units are compared in this section: Deconvolution up-sampling unit (Deconv) [75] and U-net up-sampling unit (UnetDec) [87]. Deconv only includes a 3×3 deconvolution layer to up-sample the feature maps. UnetDec consists of a 3×3 deconvolution layer followed by two successive 3×3 convolution layers. All the convolution and deconvolution layers are followed by batch normalization and ReLU activation.

We integrate the three up-sampling variants with the three feature combination methods, resulting in nine architectures. Following [87, 127], for the architectures using addition or concatenation as the feature combination unit, the corresponding up-sampling unit halves the number of the feature channels. For the architectures

with attention layers, we make the output of all the three up-sampling units be 256-d feature maps as discussed in Table 5.1.

Figure 5.8 shows the training behavior on LPBA40, micro-average accuracy is used to measure the performance of the nine architectures on the training data and the validation data. By comparing the three up-sampling units, we found that all of them achieve comparable performance at 150th epoch on the training data when the same feature combination method is applied. However, the UnetDec converges slower than the other two. On the validation data, The Resblock outperforms the other two up-sampling units, especially when the attention model is employed as the feature combination. The UnetDec achieves similar performance to the Deconv in combination with attention model whereas its validation accuracy is much lower than the other two when addition or concatenation are employed.

By comparing the training curves of three feature combination units, we did not find any obvious difference for the training accuracy. However, the attention model (Figure 5.8b) achieves higher validation accuracy than addition and concatenation (Figures 5.8d and 5.8f). Specifically, Attention+ResBlock achieves the highest accuracy among all the nine architectures, while Addition+UnetDec acquires the lowest accuracy on the validation data.

Table 5.3 summarizes the mean Dice coefficients, number of the parameters, and the inter time for the nine architectures. The mean Dice coefficients are evaluated on the model of the 150th epoch on training and validation data; the parameters of the encoder net are not counted in Table 5.3. The Attn+ResBlock achieves the highest validation segmentation score. In contrast, Concat+Deconv obtains the highest training Dice coefficient while a low validation Dice coefficient. Table 5.3 demonstrates that addition and concatenation achieve higher training accuracy but a lower validation accuracy compared with the proposed spatial attention model, which indicates that the overfitting problem exists in the architectures with attention and concatenation. Although the overfitting problem is unavoidable when the training data is

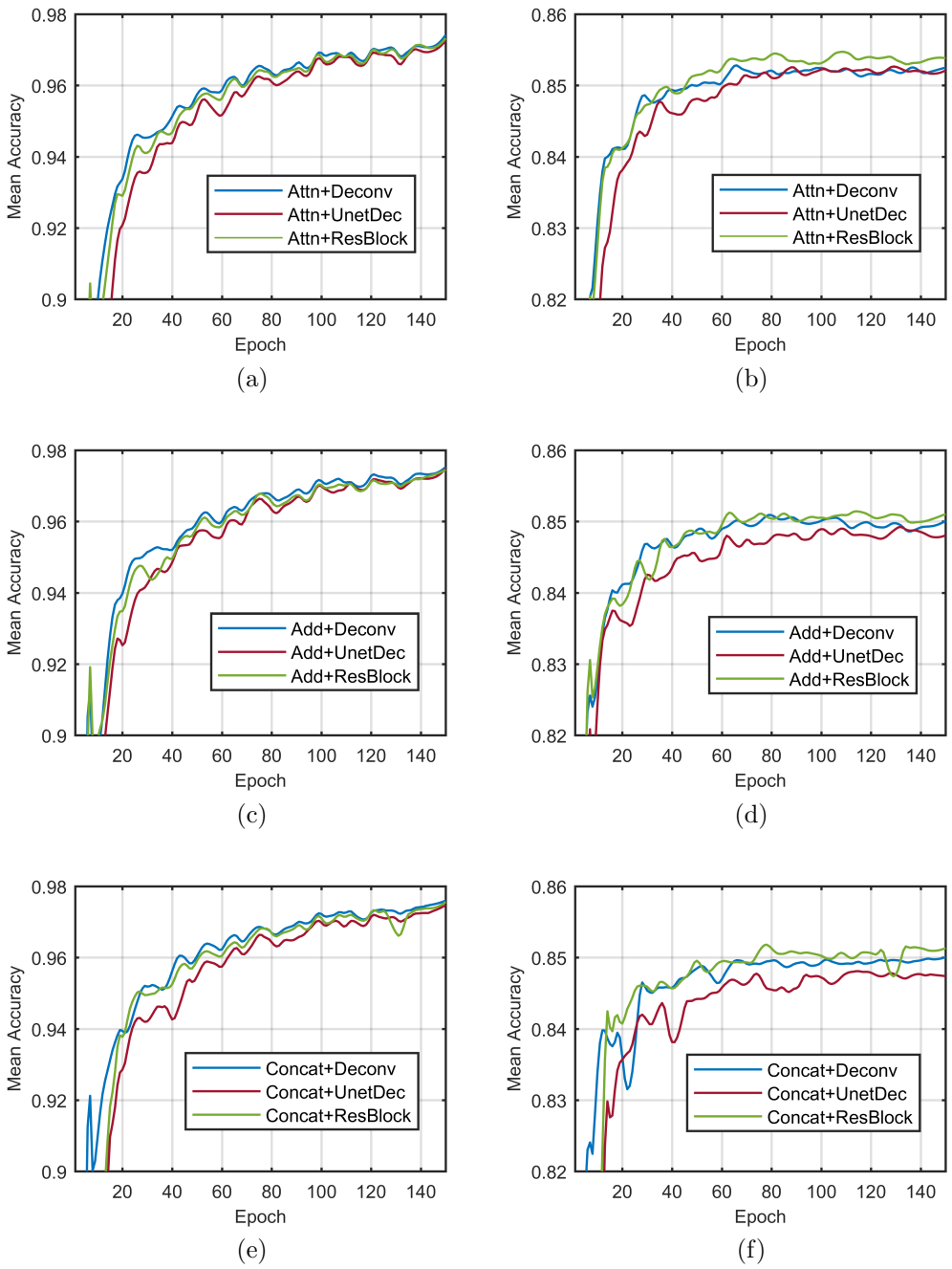


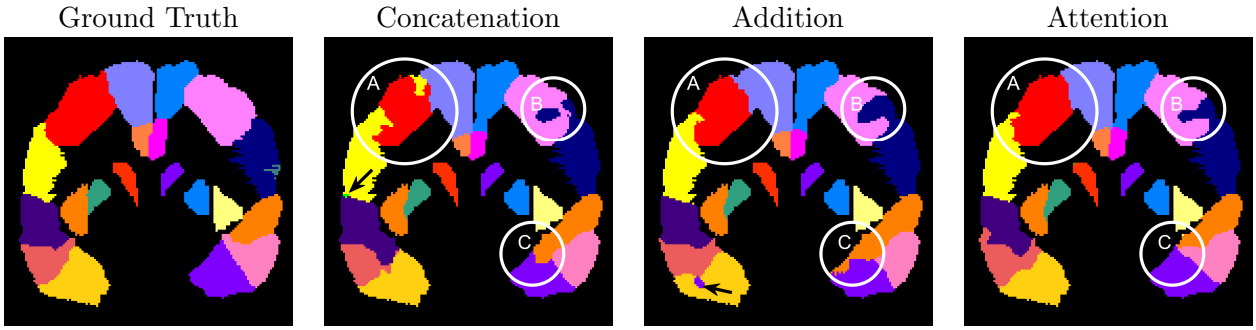
Figure 5.8: Training behavior of nine architectures on training data (left column) and validation data (right column). The nine architectures share the same encoder net and vary the decoder net and the feature combination unit. The up-sampling units are residual upsampling block (ResBlock), deconvolution up-sampling unit (Deconv), and U-net up-sampling unit (UnetDec). Also, the feature combination units are spatial attention model (Attn), addition (Add), and concatenation (Concat).

Table 5.3: Dice coefficients, parameter numbers, and the inference time of 2D slice of nine architectures with different up-sampling variants and feature combination variants. On validation data, the Attn+ResBlock achieves a significant improvement over the other eight nets in validation Dice coefficients, according to a paired, two-sided t-test ($p < 0.001$).

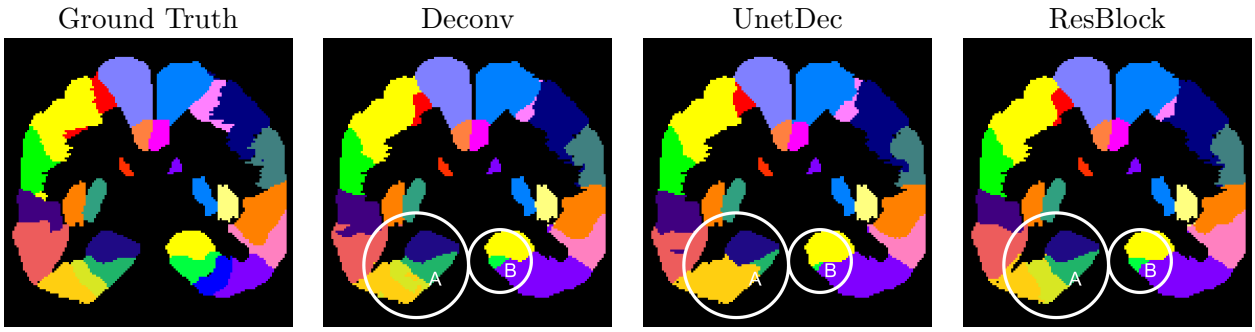
	Params (M)	Train Dice	Validation Dice	Infer time (ms)
Attention+Deconv	6.80	0.974 ± 0.001	0.860 ± 0.011	24.5
Attention+UnetDec	11.53	0.973 ± 0.002	0.855 ± 0.011	29.6
Attention+ResBlock	7.08	0.974 ± 0.001	$\mathbf{0.864 \pm 0.010}$	28.4
Addition+Deconv	7.08	0.978 ± 0.001	0.851 ± 0.010	19.2
Addtion+UnetDec	14.56	0.973 ± 0.002	0.848 ± 0.011	21.3
Addition+ResBlock	9.25	0.974 ± 0.001	0.853 ± 0.011	20.9
Concatenation+Deconv	9.21	$\mathbf{0.979 \pm 0.001}$	0.852 ± 0.010	19.8
Concatenation+UnetDec	16.45	0.977 ± 0.001	0.845 ± 0.011	21.5
Concatenation+ResBlock	10.45	0.977 ± 0.001	0.853 ± 0.001	20.1

limited, the spatial attention model effectively reduces the depth of the combined feature maps to a small number and has less trainable parameters compared with the other combination techniques, which contributes to the alleviation of the overfitting problem. Models using the spatial model have much fewer parameters than their counterparts using addition or concatenation. However, the inference time of the AttentionNet is longer than the other two feature combination methods because the AttentionNet introduces additional matrix manipulation to compute the aggregated feature maps.

Figure 5.9 demonstrates the visual quality comparisons with regard to the feature combination variants and up-sampling unit variants. Figure 5.9a illustrates the segmentation results of three feature combination units using ResBlock as the up-sampling unit, indicating that the attention model generally yields more accurate segmentation (e.g., area A and C). Moreover, since the attention model aggregated relevant features from the neighborhood, the predictions do not demonstrate isolated regions compared with the segmentation results using addition and concatenation (the isolated regions are indicated by arrows). Figure 5.9b shows examples of different up-sampling units using the spatial attention model as the feature combination unit. The ResBlock outperforms the other two up-sampling units in predicting the



(a)



(b)

Figure 5.9: Visual quality comparison of (a) different feature combination methods, (b) different up-sampling units. In (a), area A and C show that the attention outperforms concatenation and addition in predicting the details; area B demonstrates the common mistake made by the three methods; arrows refer to the "isolated regions" predicted by the concatenation and addition. In (b), area A illustrates that ResBlock yields better segmentation performance; area B is the common mistake of the three up-sampling units.

details. For example, most of the structures in area A are mislabeled for UnetDec, while the segmentation of area A produced by Deconv demonstrates intensive label inconsistency. In contrast, ResBlock successfully predicts the structure details in area A. However, due to the complexity of the brain anatomical structure, it is difficult to find a model that performs very well on all the anatomical structures of the brain. Area B in Figure 5.9a and area B in Figure 5.9b show common mistakes.

Table 5.4: Comparison of encoder nets on LPBA40 dataset, including the parameters of the encoder, mean Dice coefficients on validation data, and inference time per coronal slice. The 8x up-sampling scheme in FCN (FCN-8s) is used as the baseline.

Encoder	Params(M)	FCN-8s (baseline)		AttentionNet	
		Dice	Time (ms)	Dice (pre-trained)	Tme (ms)
ResNet-50	23.56	0.852 ± 0.010	16.0	0.864 ± 0.010	28.4
ResNet-101	42.61	0.853 ± 0.011	22.2	0.863 ± 0.011	34.5
DenseNet-121	70.33	0.853 ± 0.010	22.5	0.866 ± 0.011	34.7
VGG-16	14.71	0.841 ± 0.011	20.5	0.850 ± 0.011	31.7

5.3.4 Integration with Modern Classification Nets

We also investigate the performance of integrating the AttentionNet with different modern classification nets on LPBA40 dataset. In this section, we employ four state-of-the-art encoder networks, including ResNet-50 [41], ResNet-101 [41], DenseNet-121 [47], and VGG-16 [101] (only the convolution layers are used for VGG-16), and integrate them with the proposed AttentionNet. All the nets are trained on the corresponding pre-trained model using the training scheme as described in Section 5.3.2. For comparison, we use the 8x up-sampling scheme in FCN (FCN-8s) [75] as the baseline.

Table 5.4 reports the number of the parameters of the encoder net, validation Dice coefficients, and corresponding inference time of the 2D coronal slice. For each encoder net, the corresponding AttentionNet yields a significant improvement over the FCN-8s, according to paired, two-sided t-tests ($p < 0.001$). Among the four encoder nets, the mean Dice coefficient of VGG16 is not comparable with the other three encoder nets. However, we do not see any significant difference in the mean Dice coefficients of Attn-ResNet-50, Attn-ResNet-101, and Attn-DenseNet-121. For a well-performed model, the number of trainable parameters relates to the amount of the training data. As shown in Table 5.4, the trainable parameters of ResNet-101 and DenseNet-121 are too many to train on a small or medium dataset like LPBA40. As a result, increasing the trainable parameters contributes to the improvement of training accuracy other than the validation accuracy, which is the overfitting problem.

5.3.5 Comparison with the State-of-the-art Architectures

In this section, we compare the Attn-ResNet-50 with another two state-of-the-art 3D CNN architectures, V-net [80] and Deepmedic [61], which have been successfully applied to medical image segmentation. V-net is a 3D encoder-decoder like network. Following the training scheme in [80], we cropped the inputs to $128 \times 128 \times 64$ and trained the network with the batch size of 2. Deepmedic [61] is a fully convolutional network which is trained on image segments. We followed the training scheme in [61] and trained Deepmedic on image segments of size $37 \times 37 \times 37$ with mini-batches of 10. Since V-net and Deepmedic are trained from the scratch, we trained the two 3D models for 300 epochs with an initial learning rate of 0.01. Like the Attn-ResNet-50, we adopt the Adam optimizer and applied exponential decay to the learning rate. Attn-ResNet-50 is trained using the training scheme discussed in Section 5.3.2. We exam the performance of the three networks on IBSR dataset.

The mean Dice coefficients of the validation set are summarized in Table 5.5. The proposed AttentionNet demonstrates significant improvement compared with the other two 3D CNN architectures, according to paired, two-sided t-tests ($p < 0.001$). Figure 5.10 lists segmentation results of three coronal slices in the IBSR dataset. Compared with the 3D networks, Attn-ResNet-50 produces a precise segmentation (e.g., area B and C in slice 1). Moreover, Attn-ResNet-50 outperforms the V-net and Deepmedic in the segmentation of anatomical structures with small size (e.g., area A and B in slice 2). On the other hand, for the regions with fewer labels (e.g., slice 3), V-net and Deepmedic achieve comparable segmentation performance with Attn-ResNet-50. For the 3D networks, since the training samples are cropped into small volumetric images, it is hard to capture the large contextual relation, which hinders the ability to predict details, especially for the complicated anatomical structures in the brain. For V-net and Deepmedic, because of the limitation of memory, we adopt overlap-tile strategy [87] for inference stage to obtain seamless segmentation, which leads to a slower inference speed.

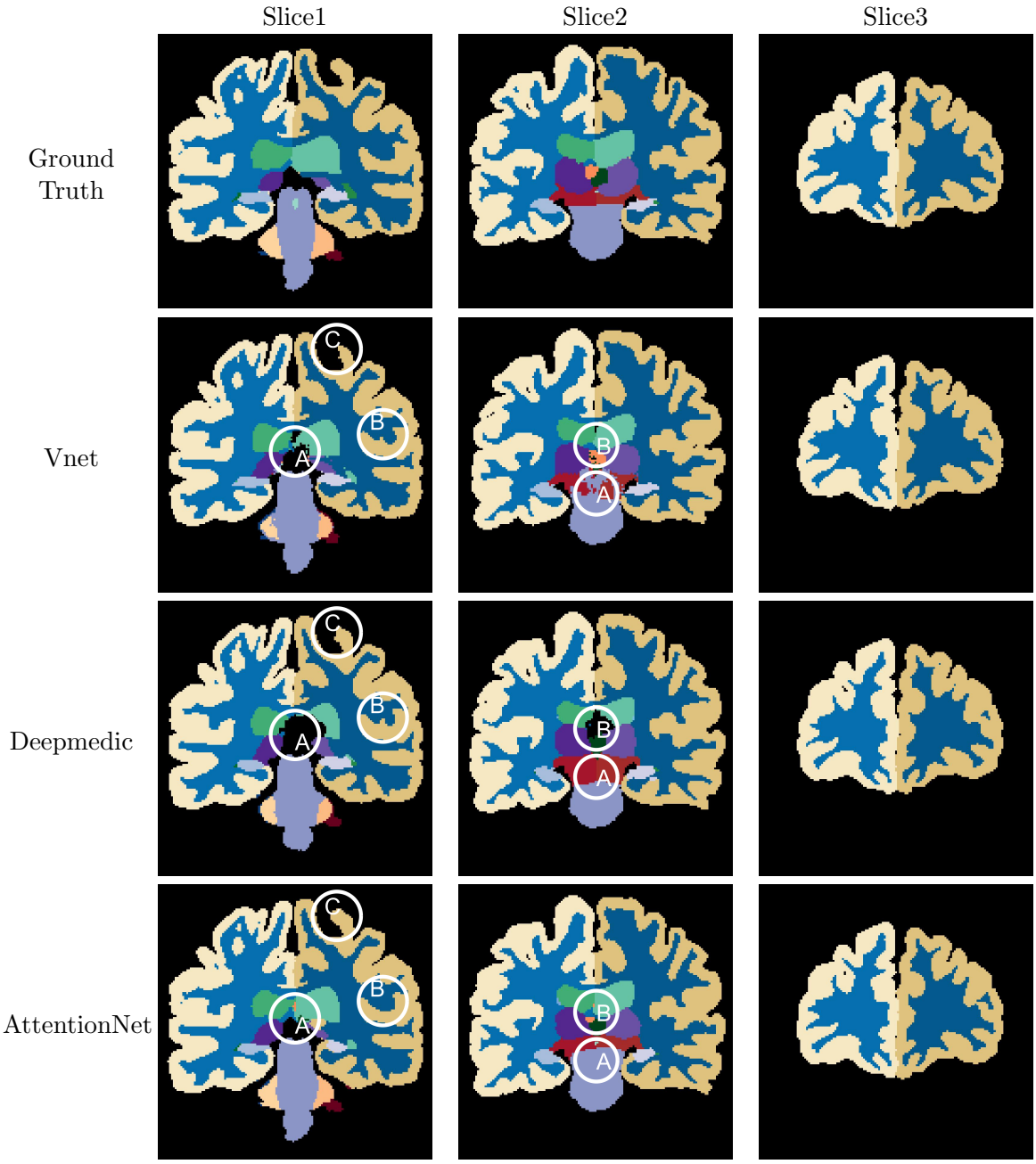


Figure 5.10: Examples of segmentation results on the IBSR dataset.

Table 5.5: Validation Dice coefficients and inference time of each 3D image of the proposed AttentionNet with the state-of-the-art architectures on IBSR.

Network	Dice	Infer time(s)
Attn-Resnet50	0.840 ± 0.022	3.4
V-Net	0.725 ± 0.040	16
DeepMedic	0.803 ± 0.028	36

5.4 Conclusion

In this chapter, we proposed a novel spatial attention model, which can be applied to selectively combine the finer features with the up-sampling path in the encoder-decoder architecture. The proposed attention model captures the spatial dependencies between the deep features from the up-sampling path and the finer features from the down-sampling path. By attending the relative positions, the attention model outputs weighted aggregated finer features, which contributes to the precise segmentation of the brain anatomical structure. Compared with the other feature combination methods, the proposed spatial attention model achieves outstanding performance in terms of Dice coefficients and visual segmentation results.

Moreover, the proposed 2D architecture yields significant improvement compared with the 3D architectures. The 2D architecture benefits the networks in terms of the usage of the large batches and the capture of large context information in the coronal plane. Moreover, the embedded position information also provides the net with 3D spatial prior, which is effective for the anatomical structure segmentation of brain MR images.

Chapter 6

End-to-End Trainable CNN-CRF with High Order Potentials

In Chapter 4, we have demonstrated that the superpixel based graphical model benefits the brain anatomical structure segmentation in terms of encouraging the labeling consistency. In order to combine the strength of the CNN with that of the graphical model, we propose an end-to-end network to incorporate CNN with high order CRF of which the high order term is defined on the superpixels. Moreover, we derive the mean field inference for the proposed high order CRF, which enables to implement the CRF inference as a stack of the CNN building layers. Therefore, the errors can be back-propagated to the CNN layers through the CRF layers so that the parameters of the CNN and CRF can be learned jointly. Extensive experiments demonstrate that the proposed end-to-end trainable network significantly improves the segmentation performance compared with the existing methods that combine the CNN with the CRF. Moreover, we also experimentally show that the proposed high order CRF can be integrated with the modern CNN models and improve the segmentation performance quantitatively and qualitatively.

6.1 Introduction

In Chapter 4, we have demonstrated that employing a graphical model yields a significant improvement of the segmentation accuracy for brain anatomical structure segmentation. The smoothness term associated with the graphical model encourages the labeling consistency between neighboring pixels and alleviates the labeling inconsistency produced by the unary classifier. Recently, CNN has achieved remarkable improvements in both semantic and medical image segmentation. Although the large receptive fields contribute to the capture of rich contextual information, the inconsistent labeling problem still exists in the CNN based segmentation models. Therefore, we propose to combine the strengths of the graphical model with the state-of-the-art CNN architecture, aiming at addressing the label inconsistency problem and improving the segmentation performance.

Several studies have been proposed to combine the CNN architecture with the graphical model (e.g., CRF). One of them [23, 61] employs CRF as a post-processing strategy to refine the labeling results obtained by CNN. Although the post-processing strategy contributes to the quantitative and qualitative improvements, the separate training system disables the CNN to adapt its weights to the CRF during the training phase [131]. Alternatively, several frameworks are developed to incorporate the CRF into the deep networks so that they can be trained jointly. Lin et al. [72] model the pairwise potentials as the fully connected layers. However, this modeling formulation of the pairwise potentials outputs $L \times L$ channels that indicate the possible label combinations for a pair of pixels, which is expected to consume very high memory for training. Liu et al. [73] employ the mean field algorithm to obtain the CRF inference, and thus model the pairwise potentials as a stack of well-designed convolution layers.

Most of the existing methods that combine CRF with CNN [23, 61, 72, 73, 131], no matter the post-processing methods or the end-to-end trainable models, employ the pairwise CRF (refer Chapter 2 for details). However, high order potentials, which also play an essential role in improving the segmentation accuracy, are rarely discussed in

the existing studies. Besides, experiments in [73, 131] demonstrated that end-to-end training of CNN and CRF yields a significant improvement over the disjoint training system. In this chapter, we aim at developing an end-to-end trainable network that combines CNN with the high order CRF.

Like Chapter 4, we define the high order potentials over the superpixels to encourage the labeling consistency in the superpixel. We solve the high order CRF using the mean field approximation and model each iteration of the mean field algorithm as a stack of the building layers in CNN. Extensive experiments indicate that involving the superpixel based high order CRF contributes to the improvement of the segmentation accuracy for the brain anatomical structure segmentation. The contributions of this work include:

1. We derive the mean field approximation of the superpixel-based high order potential and model the mean field update of the high order potential as building blocks in CNN.

2. We develop a semi-dense pairwise potential and model the corresponding mean field update as a depthwise convolution.

3. We obtain the superpixel segments based on the predicted label images and update the superpixel segments during the training stage in order to guarantee the positive labeling consistency in superpixels.

This chapter is organized as follows. In Section 6.2, we derive the formulation of the mean field approximation for the high order CRF inference, and describe the details of modeling the mean field update as CNN building blocks. Then we perform the network analysis in Section 6.3, and the performance of combining the proposed high order CRF with different state-of-the-art CNNs are evaluated in the same section.

6.2 Method

6.2.1 CRF with High Order Potentials

Given an image I, a random field is defined over a set of random variables $\mathbf{X} = \{X_1, X_2, \ldots, X_N\}$, where each random variable is associated with a corresponding image pixel $i \in \{1, 2, \ldots, N\}$ and takes a value from the label set $\mathcal{L} = \{l_1, l_2 \ldots, L\}$. The CRF energy of the configuration $x \in \mathcal{L}^N$ is

$$E(x) = \sum_{i \in \mathcal{V}} \psi_u(x_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_p(x_i, x_j) + \sum_{s \in \mathcal{S}} \psi_h(x_s) \tag{6.1}$$

where $\mathcal{V}$ denotes the set of the pixels in the image, $\mathcal{N}_i$ is the neighborhood of pixel i, and $\mathcal{S}$ denotes the set of high order cliques, i.e., superpixels in this work. In the energy function, $\psi_u(x_i)$, $\psi_p(x_i, x_j)$, and $\psi_h(x_s)$ represent unary potential, pairwise potential, and high order potential, respectively.

The **unary potential** $\psi_u(x_i)$ is defined on unary pixels, and it describes the cost of assigning the label x_i to the pixel at $i \in \mathcal{V}$. In this model, the unary potential is computed as the negative log likelihood of pixel i taking the label, x_i, which is produced by the CNN.

The **pairwise potential** $\psi_p(x_i, x_j)$ measures the cost of assigning a pair of labels x_i, x_j to a pair of pixels at i, j. Krähenbühl et al. [68] demonstrate that densely connected system, where the CRF is defined over a complete graph (refer to Section 2.3 for details), leads to significant improvements for semantic segmentation. However, for brain anatomical structure segmentation, the interaction between two pixels can be neglect if the distance measure is very large (e.g., pixels in the left hemisphere and right hemisphere). Consequently, instead of the fully connected system [68] or grid neighborhood system [99], we propose a *semi-dense pairwise potential* which is defined over the neighborhood system $\mathcal{N}$, where $\mathcal{N}_i$ denotes a $r \times r$ neighborhood of the pixel i.

Following [68], we use the Gaussian function to model the pairwise potentials:

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) \exp\left(- \frac{|p_i - p_j|^2}{2\theta_\gamma^2} \right) \tag{6.2}$$

The Gaussian kernel, which is defined on the spatial location p, encourages the nearby pixels to be assigned the same label. θ_γ is the standard derivation of the Gaussian kernel, which will be discussed in Section 6.2.2. The compatibility function $\mu(x_i, x_j)$ captures the label compatibility between two nearby pixels. It introduces a penalty for assigning different labels to a pair of pixels. If the two labels are highly compatible, the compatibility function assigns a low penalty for this pair of labels. For example, "superior frontal gyrus" indicates that nearby pixels may be "middle frontal gyrus" in the anatomy of the human brain, thus the label compatibility of "superior frontal gyrus" and "middle frontal gyrus" is high. Otherwise, a high penalty is assigned to the pair of labels. The $\mu(x_i, x_j)$ is learned from the data, as described in Section 6.2.2.

The **high order potential** $\psi_h(x_s)$ measures the cost of the label assignment $x_s = \{x_i : i \in s\}$ for a high order clique s which is associated with a set of pixels (refer to Section 2.3, for details). In this study, we use the superpixel-based high order potential, where the high order cliques refer to superpixels. The P^N Potts is employed to model the high order potential, encouraging the pixels inside the superpixel to take the same label:

$$\psi_h(x_s) = \begin{cases} w_h \gamma^l, & \text{if } \forall i \in S, x_i = l \\ w_h \gamma^{max} & \text{otherwise} \end{cases} \tag{6.3}$$

The P^N Potts penalizes the case of inconsistent labeling throughout the superpixel with a high cost γ^{max}, otherwise obtains a low cost γ^l if all the pixels in the superpixel are assigned the same label. w_h is the learnable parameter, indicating the weight of the high order potential. The configurations of γ^{max}, γ^l, and w_h are described in Section 6.2.2

6.2.2 Mean Field approximation of the high order CRF

Mean field approximation algorithm [66] is employed to approximate the posterior probability $P(\mathbf{X} \mid \mathbf{I})$ of the CRF. The mean field algorithm computes a distribution

$Q(\mathbf{X})$ to approximate $P(\mathbf{X})$ by minimizing the KL-divergence. $Q(\mathbf{X})$ is a simpler distribution, which can be expressed as the product of the independent marginals:

$$Q(\mathbf{X}) = \prod_i Q_i(X_i) \tag{6.4}$$

Minimizing the KL-divergence yields the general iterative update equation of the mean field inference [66]:

$$Q_i(x_i = l) = \frac{1}{Z_i} \exp\Bigg(-\sum_{c \in \mathcal{C}} \sum_{x_c | x_i = l} Q_{c-i}(x_{c-i}) \psi_c(x_c) \Bigg) \tag{6.5}$$

where Z_i is the partition function, x_c is the label assignment of the clique c, and x_{c-i} is the label assignment of the clique apart from pixel i. Equation (6.5) indicates that the mean field update of variable X_i is the summation of the mean field updates of all the cliques associated with variable X_i. In our model, each variable X_i is associated with three types of cliques: unary clique, pairwise clique, and high order clique (superpixel). Consequently, to obtain the mean field for the proposed high order CRF, the mean field update for each clique is a prerequisite.

Based on the assumption in Equation (6.4), we obtain that $Q_{c-i}(x_{c-i}) = \prod_{j \in c, j \neq i} Q_j(x_j)$. For the high order potential, by substituting Equation (6.3) into the general update equation in Equation (6.5), we obtain the mean field update of the high order potential:

$$\sum_{x_s | x_i = l} Q_{s-i}(x_{s-i}) \psi_h(x_s) = \prod_{j \in s, j \neq i} Q_j(x_j = l) w_h \gamma^l + (1 - \prod_{j \in s, j \neq i} Q_j(x_j = l)) w_h \gamma^{max} \tag{6.6}$$

Similarly, the update of the pairwise clique is derived by putting Equation (6.2) into Equation (6.5), which is:

$$\sum_{j \in \mathcal{N}_i} \sum_{x_j | x_i = l} Q_j(x_j = l') \psi_p(p_i, p_j) = \sum_{l' \in \mathcal{L}} \mu(l, l') \sum_{j \in \mathcal{N}_i} Q_j(x_j = l') k_G(p_i, p_j) \tag{6.7}$$

where $k_G(p_i, p_j)$ denotes the Gaussian kernel in Equation (6.2). The update of the

mean field inference for the proposed high order CRF is:

$$
\begin{aligned}
Q_i(x_i = l) = & \frac{1}{Z_i} \exp\Bigg(- \psi_u(x_i) - \sum_{l' \in \mathcal{L}} \mu(l, l') \sum_{j \in \mathcal{N}_i, j \neq i} Q_j(x_j = l') k_G(p_i, p_j) \\
& \prod_{j \in s, j \neq i} Q_j(x_j = l) w_h \gamma^l + (1 - \prod_{j \in s, j \neq i} Q_j(x_j = l)) w_h \gamma^{max} \Bigg)
\end{aligned}
\tag{6.8}
$$

Equation (6.8) results in the following algorithm:

Algorithm 1: Mean field of the high order CRF

Result: The approximated distribution $Q(\mathbf{X})$

Initialization: $Q_i(x_i) \leftarrow \frac{1}{Z_i} \exp(-\psi_u(x_i))$;

while *not converged* **do**

- Pairwise message passing: $\tilde{Q}_i(l) \leftarrow \sum_{j \in \mathcal{N}_i, j \neq i} k_G(p_i, p_j) Q_j(l)$;
- Compatibility transform: $\hat{Q}_i(x_i) \leftarrow \sum_{l \in \mathcal{L}} \mu(x_i, l) \tilde{Q}_i(l)$;
- High order message passing: $\check{Q}_i(x_i) \leftarrow w_h \Big(\prod_{j \in s, j \neq i} Q_j(l) \gamma^l + (1 - \prod_{j \in s, j \neq i} Q_j(l)) \gamma^{max} \Big)$;
- Local update: $\breve{Q}_i(x_i) \leftarrow -\psi_u(x_i) - \check{Q}_i(x_i) - \hat{Q}_i(x_i)$;
- Normalization: $Q_i(x_i) \leftarrow \frac{1}{Z_i} \exp \breve{Q}_i(x_i)$

end

Each iteration can be decomposed into five steps: pairwise message passing, compatibility transform, high order message passing, local update, and normalization. In the next section, we will describe the details of performing the mean field inference as a stack of CNN layers.

6.2.3 Architecture

Initialization: Following [131], we use the output logits $U(x_i)$ of the CNN as the negative unary potential, i.e., $U(x_i) = -\psi_u(x_i)$. $Q_i(x_i)$ is initialized with $\frac{1}{Z_i} \exp(-\psi_u(x_i))$, where Z_i is the partition function, $Z_i = \sum_{x_i} \exp(-\psi_u(x_i))$. Therefore, initializing $Q_i(x_i)$ is equivalent to applying a softmax function to $U(x_i)$.

Pairwise message passing: As discussed in Section 6.2.1, $\mathcal{N}_i$ indicates a $r \times r$ neighborhood of pixel i. Therefore, the pairwise message passing can be expressed as

a convolution with the Gaussian kernel k_G:

$$\tilde{Q}_i(l) = \sum_{j \in \mathcal{N}_j} k_G(p_i, p_j) * Q_j(x_i = l) - Q_i(x_i = l) \tag{6.9}$$

$Q_i(x_i = l)$ is subtracted from the convolution because the message passing does not sum over Q_i. Since the spatial location is fixed, we can precompute the Gaussian kernel and perform the convolution on each channel of Q_i. Therefore, the pairwise message passing can be implemented as a $r \times r$ depthwise convolution layer followed by subtracting Q_i from the convolution results without learnable parameters for this step.

Compatibility transform: The compatibility transform can be implemented as a 1×1 convolution, and the inputs and outputs channels are both the number of the classes, L.

High order message passing: In the high order message passing step, we first compute the joint probability $\prod_{j \in s} Q_j(x_j = l)$. Then $\prod_{j \in s, j \neq i} Q_j(x_j = l)$ can be computed as dividing $\prod_{j \in s} Q_j(x_j = l)$ by $Q_i(x_i = l)$. We set γ^l to be the negative log of the average of the Q values across all the pixels within the superpixel, i.e., $\gamma^l = -\log \frac{1}{N_s} \sum_{j \in s} Q_j(x_j = l)$ with N_s being the number of pixels within the superpixel, and multiply γ^l by $\prod_{j \in s, j \neq i} Q_j(x_j = l)$ elementwise. On the other hand, we use independent γ^{max} for each class. As a result, $(1 - \prod_{j \in s, j \neq i} Q_j(l))\gamma^{max}$ is equivalent to stacking a scale layer on the results of $1 - \prod_{j \in s, j \neq i} Q_j(l)$, where γ^{max} with size of $1 \times 1 \times 1 \times L$ serves as the parameter of the scale layer.

Last, motivated by the intuition that the importance of high order potential for each class is different, we use class-specific weight w_h for the high order potential. Therefore, multiplying w_h is equivalent to stacking a 1×1 convolution layer, with inputs and outputs channels being the number of the classes, L.

Local update: In this step, the output of unary update $\psi_u(x_i)$, pairwise update $\hat{Q}_i(x_i)$, and high order update $\breve{Q}_i(x_i)$ are summed up elementwise.

Normalization: Based on the definition of the partition function, the normalization

can be implemented using a softmax layer.

One iteration of the mean field approximation of the proposed high order CRF is shown in Figure 6.1. The inputs include the spatial positions p, the superpixel segments S, the negative unary potential U, and the unnormalized distribution at the tth iteration $\breve{Q}^t$. For each iteration, we start with normalization so that the initialization can be integrated into the building block. The output $\breve{Q}^{t+1}$ is the unnormalized distribution at $t+1$ iteration.

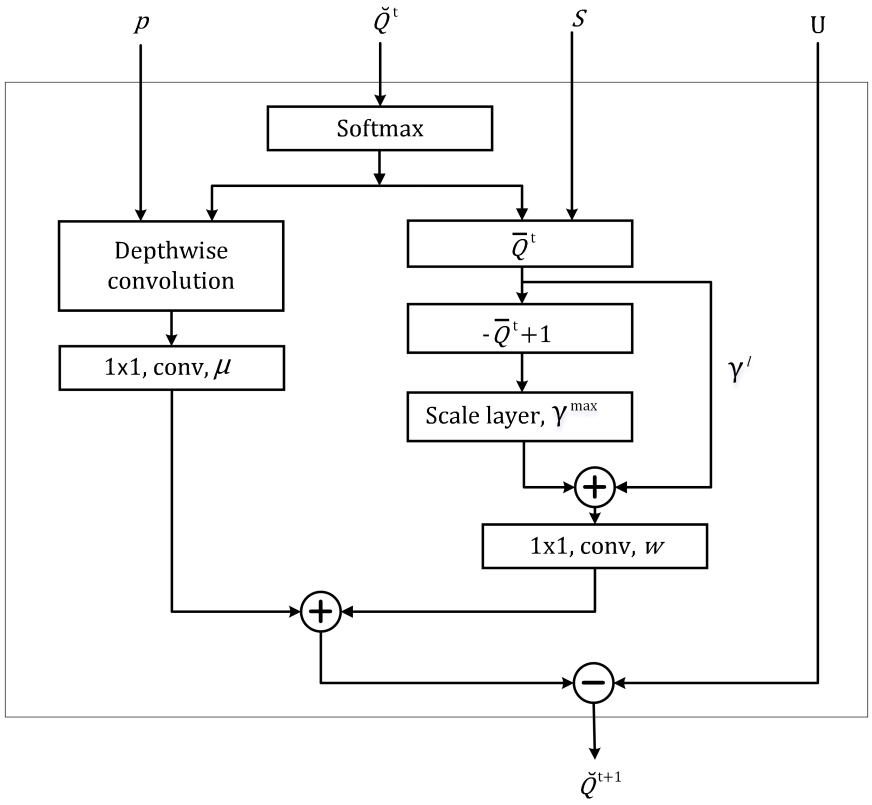


Figure 6.1: Building block of one iteration of the mean field approximation algorithm, $\bar{Q} = \prod_{j \in c, j \neq i} Q_j(x_j = l)$.

The architecture of the unified network that combines CNN with the proposed high order potential (CNN-HOCRF) is shown in Figure 6.2. The iterative update of the mean field approximation is implemented by stacking several mean field building blocks shown in Figure 6.1. The number of the mean field building block refers to the iterations of the algorithm, which is a hyperparameter in this network.

As illustrated in Figure 6.2, the four inputs of the CRF are obtained as follows. (1) U is the raw scores (a.k.a logits) obtained from the CNN, which represents the negative unary potential $-\psi_u$. (2) For the superpixel segmentation, we apply SLIC algorithm [1] on the estimated label image (a.k.a pixelwise predictions obtained from the CNN) instead of the intensity image, as discussed in Chapter 4. Specifically, the superpixel segmentation is updated and refined during the training stage, resulting in the encouragement of the positive labeling consistency in the superpixel. (3) To perform the pairwise message passing, we precompute the Gaussian kernel k_G based on the spatial positions p and the size of $\mathcal{N}_i$. (4) According to Algorithm 1, $-\psi_u$, i.e. U, is used to initialize $\breve{Q}^0$, while for the rest mean field building blocks, the input $\breve{Q}^t$ is the output of the previous mean field building block. The CRF parameters share the same values for each mean field approximation block.

Since the mean field building block outputs the unnormalized Q, we stack a softmax layer after the last mean field building block to produce the probability maps that sum to one. Since each step in Algorithm 1 can be implemented as the basic building layer or operation in CNN, error derivatives w.r.t. the parameters can be calculated during back-propagation, resulting in an end-to-end trainable network.

For the CNN part, we adopt a 2D CNN architecture which benefits the model in terms of low memory requirement, deep architecture, and fine-tuning on pre-trained mode.

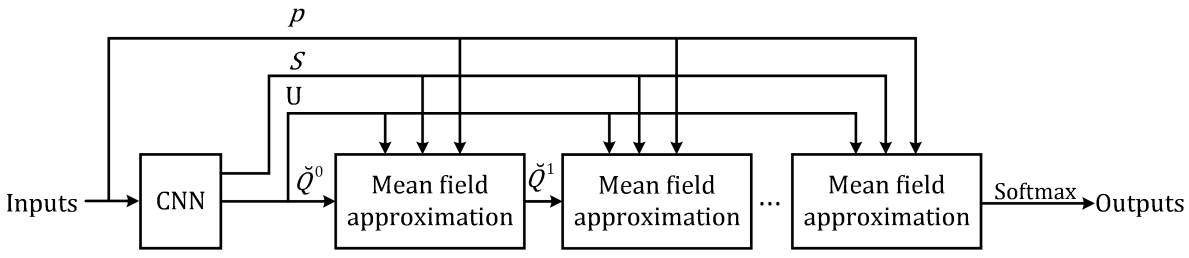


Figure 6.2: Architecture of the proposed CNN-HOCRF.

6.3 Experiments

6.3.1 Preprocessing

First, decile intensity normalization is applied to the 3D images to ensure the tissues to have the similar intensity distribution. Then, the 3D data are sliced into 2D coronal images. Last, after adding the relative coordinates, all the input images are normalized to have zero mean by subtracting the mean value.

6.3.2 Implementation Details

A two-stage training strategy is applied to train the network. First, the CNN part is trained alone at the first stage. Then, the high order CRF is added to the network, and all the parameters are learned jointly. The network is trained using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. We train the CNN part with an initial learning rate of 0.001 for 50 epochs while we fine-tune the whole net with an initial learning rate of 0.0005 for another 100 epochs. Exponential decay is applied to the learning rate, where the learning rate decays every 25 epochs with a base of 0.8. Online data augmentation with random scaling, random rotation, and random cropping is applied before training. As demonstrated in Figure 6.2, the superpixel segmentation S is obtained by performing SLIC on the outputs of CNN. To improve computational efficiency, we update the superpixel segmentation every 10 epochs during the second training stage. All the networks are evaluated with 5-fold cross-validation on the corresponding datasets. Our CNN is implemented using the Tensorflow library on an NVIDIA GTX 1080 Ti GPU.

For the parameters of the high order CRF, we initialize the compatibility matrix μ with the Potts model, which is -1 on the diagonal and 0 elsewhere. γ^{max} is initialized with $-\log(0.005)$, truncating the high order cost if the probability of superpixel taking label l is lower than 0.005. The high order weight is initialized with an identity matrix, which is 1 on the diagonal and 0 elsewhere. The standard deviation of the Gaussian

kernel θ_γ is obtained from a cross-validation process. Dice coefficient between the ground truth and the dense predictions is used as the evaluation metric.

6.3.3 Evaluation of the Hyperparameter

All the models examined in this section use FCN-8s as the CNN part. We report the mean validation Dice coefficients of the LPBA40 dataset with a 5-fold cross-validation. All the coronal images are center cropped to the size of 161×161. The parameters of the CNN are initialized with the ImageNet pre-trained model.

6.3.3.1 Number of the mean field iterations

Each mean field approximation building block models one iteration in the mean field algorithm. We examine the mean Dice coefficients with respect to the number of the mean field approximation building blocks. In this experiment, the neighborhood $\mathcal{N}_i$ that defines the pairwise potential is set to 5×5. Figure 6.3 illustrates that the proposed model reaches a good segmentation accuracy by stacking three mean field approximation building blocks. It is notable that the end-to-end trainable CNN-CRF [131] needs 5 to 10 iterations to converge while CRF post-processing [68] requires 10 to 20 iterations to converge. In the rest experiments, we stack three mean field approximation blocks for the CNN-HOCRF network.

6.3.3.2 Approximate size of the superpixel

SLIC algorithm is a modified k-means clustering method, and its superpixel segmentation relates to the parameter k which approximates the number of superpixel segments. For the brain anatomical structures of which the size varies from one to another, coarse superpixel segmentation leads to an increase in the number of incorrectly clustered superpixels while fine superpixel segmentation decreases the computational efficiency. In this section, we study the effect of varying the parameter k in the SLIC. All the models examined in this experiment adopt 5×5 neighborhood for the pairwise

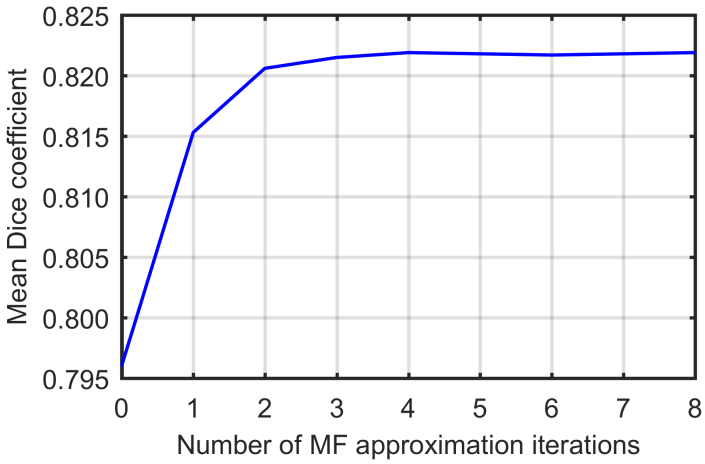


Figure 6.3: Dice coefficients w.r.t. the number of mean field iterations

Table 6.1: Mean Dice coefficients of different settings of the average size of the superpixel (SP).

SP Size (k)	9×9 (320)	8×8 (400)	7×7 (530)	6×6 (720)	5×5 (1000)
Dice	0.805 ± 0.011	0.816 ± 0.011	0.821 ± 0.011	0.822 ± 0.011	0.822 ± 0.011

potential.

For the inputs with size of 161×161, we compared different settings of the average size of the superpixel, including '9×9','8×8','7×7','6×6', and '5×5' (the corresponding k are 320, 400, 530, 720, and 1000, respectively). Table 6.1 demonstrates that the '6×6' and '5×5' achieve the best mean Dice coefficients, which are slightly higher than '7×7'. Although the high order CRF does not enforce the pixels inside the superpixel to take the same label, adopting a large superpixel size (e.g., '9×9' and '8×8') tends to incorrectly encourage labeling consistency for the pixels of different anatomical structures, resulting in the decrease of segmentation performance.

6.3.3.3 Size of neighborhood in the pairwise term

Since the pairwise message passing is implemented as a depthwise convolution, the size of neighborhood $\mathcal{N}_i$ relates to the receptive field. We evaluate the mean Dice coefficients of different receptive fields, including '3×3', '7×7', '15×15', and '31×31'.

Table 6.2: Mean Dice coefficients of the different sizes of the neighborhood $\mathcal{N}_i$.

Receptive field	3×3	7×7	15×15	31×31
Dice	0.821 ± 0.011	0.827 ± 0.010	0.831 ± 0.009	0.824 ± 0.011

All the models use the hyperparameter $k = 530$ for SLIC algorithm. As summarized in Table 6.2, the receptive field 15×15 achieves the highest mean Dice coefficient, which indicates that for a 161×161 image, a neighborhood of 15×15 is enough to capture the relation between pixels.

6.3.4 Ablation Study

To study the behavior of the proposed CNN-HOCRF, we conducted several ablation studies. For the proposed CNN-HOCRF, we adopt the optimal hyperparameters discussed in Section 6.3.3. First, we show the effect of employing the high order potential by comparing the proposed CNN-HOCRF with CNN-pairwiseCRF. For the pairwiseCRF, we employ the same pairwise potential model in Equation (6.2) and set the kernel size to 15×15. Next, to investigate the effect of the superpixel segmentation scheme of the proposed model, we exam the performance of implementing the superpixel segmentation on the intensity image. To this end, we adopt the same high order CRF model except that the superpixel segmentation is obtained through the intensity image (CNN-HOCRF-Intensity) using SLIC (k=530). The CNN-HOCRF, the CNN-pairwiseCRF, and the CNN-HOCRF-Intensity are trained using the same two-stage training strategy, where the CNN is trained alone for 50 epochs then the CRF is added to the network at the second stage for fine-tuning of 100 epochs. The CNN (FCN-8s), which is fine-tuned for another 100 epochs after the first training stage, is used as the baseline. Moreover, a disjoint learning scheme is included for completeness, where the denseCRF [68] is employed as a post-processing of the baseline results.

Table 6.3 reports the per-class and mean Dice coefficients for the five models. The proposed high order CRF achieves significant improvement over the other four models

in the mean Dice coefficient. Table 6.3 indicates that CRF inference, including joint training and post-processing, yields improvements over the baseline. However, CRF post-processing does not back-propagate the error to the CNN, resulting in lower Dice coefficients than the counterparts of the end-to-end training networks. Next, we compare the performances of the three end-to-end trainable networks: CNN-pairwiseCRF, CNN-HOCRF-Intensity, and CNN-HOCRF. Compared with the pairwiseCRF, introducing the high order potential significantly improves the segmentation performance in terms of the per-class accuracies and the mean accuracy. In addition, by comparing the CNN-HOCRF with CNN-HOCRF-Intensity, we found that the proposed label-based superpixel segmentation benefits most of the classes. However, for structures 'putamen', 'hippocampus', 'superior temporal gyrus', and 'middle temporal gyrus', the intensity-based superpixel segmentation achieves higher Dice scores.

Figure 6.4 illustrates the segmentation results. For the baseline method, the segmentation results show much more labeling inconsistency than those of involving the CRF inference. For example, the labeling results produced by the baseline method show some small "holes", e.g., area A and C in slice 2, while we can find few small "holes" in the segmentations obtained by the methods using CRF inference. Therefore, the CRF inference contributes to the labeling consistency of the neighboring pixels.

However, for the four models using CRF inference, Figure 6.4 also illustrates their flaws in the pixelwise prediction results. First, the post-processing tends to over-smoothes the boundaries, resulting in some structures with small size "eroded" by the surrounding/adjacent structures (e.g., area A in slice1, area A and C in slice2, and area B in slice3). Next, the CNN-pariwiseCRF has the problem of over-segmentation, where the mistaken labeling area in the CNN's output is enlarged through the pairwise CRF (e.g., area B in slice2). Then, for CNN-HOCRF-Intensity, it is difficult to obtain precise superpixel segments for the structure with small size, which leads to negative encouragement of the labeling consistency. As a result, for the small structures surrounded by other anatomical structures (e.g., area A in slice1), CNN-

HOCRF-Intensity does not yield good segmentation results. The proposed CNN-HOCRF achieves the best performance in preserving the labeling consistency and avoiding over-segmentation and over-smoothing. However, for the "isolated" structures of which the surrounding pixels are labeled as background, e.g., caudate (area A) in slice3, the CNN-HOCRF-Intensity outperforms the CNN-HOCRF, which agrees with the results in Table 6.3.

6.3.5 Visualization of the Learned HOCRF Parameters

Figure 6.5a illustrates the learned label compatibilities μ, which indicates the penalty for a pair of nearby pixels that are assigned different labels. The high penalty indicates low label compatibility between nearby pixels. From Figure 6.5a, we have the following observations: 1. label pairs of the same lobe are of high compatibility. For example, the structures in the frontal lobe, including SFG, MFG, IFG, PRFG, MOrbG, and GR, have a low penalty for each other while show a high penalty for the structures outside the frontal lobe. In other words, if a pixel is labeled as one of the structures in the frontal lobe, the neighbors of the pixel also tend to be labeled as one of the structures in the frontal lobe. 2. The subcortical structures, e.g., hippocampus, putamen, and caudate, are delineated as "isolated" structures in this dataset. Since surrounding pixels are not annotated as valid label, the subcortical structures are incompatible with any other structures except the background. 3. we observe that the labels of adjacent rows have a high penalty. This is because the labels of adjacent rows in Figure 6.5a are the same structure of the left and right hemisphere (e.g., left caudate and right caudate). Through the end-to-end training, the network learns the relation between label compatibility and the spatial context for the brain MR images.

Figure 6.5b is the learned weights of the high order term w_h, where each column is the learned kernel weights for each class. Figure 6.5b indicates that the w_h learns much less contextual information than the compatibility function μ. The reason is

Table 6.3: Per-class and mean Dice coefficient comparison on the LPBA dataset, where left and right hemisphere labels are shown jointly. The proposed CNN-HOCRF yields significant improvement comparing with the other four models, according to two-sided, paired t-test on the Dice coefficient ($p < 0.001$).

Structures (Abbreviation)	Baseline	Post-Processing	CNN-pairwiseCRF	CNN-HOCRF-Intensity	CNN-HOCRF
Frontal lobe					
superior frontal gyrus (SFG)	0.866	0.874	0.887	0.885	**0.900**
middle frontal gyrus (MFG)	0.842	0.855	0.867	0.868	**0.880**
inferior frontal gyrus (IFG)	0.798	0.807	0.823	0.829	**0.836**
precentral gyrus (PrCG)	0.794	0.806	0.822	0.816	**0.837**
middle orbitofrontal gyrus (MOrbG)	0.768	0.787	0.795	0.800	**0.808**
lateral orbitofrontal gyrus (LOrbG)	0.678	0.667	0.711	0.708	**0.728**
gyrus rectus (GR)	0.788	0.804	0.806	0.810	**0.820**
Paritetal lobe					
postcentral gyrus (PoCG)	0.753	0.770	0.780	0.779	**0.795**
superior parietal gyrus (SPG)	0.805	0.822	0.828	0.830	**0.839**
supramarginal gyrus (SMG)	0.768	0.784	0.778	0.764	**0.794**
angular gyrus (AG)	0.738	0.766	0.769	0.780	**0.782**
precuneus (PCUN)	0.771	0.774	0.790	0.786	**0.803**
Occipital lobe					
superior occipital gyrus (SOG)	0.713	0.724	0.741	0.751	**0.753**
middle occipital gyrus (MOG)	0.771	0.797	0.787	0.797	**0.799**
inferior occipital gyrus (IOG)	0.781	0.803	0.796	0.803	**0.807**
cuneus (CUN)	0.781	0.778	0.814	0.803	**0.827**
Temporal lobe					
superior temporal gyrus (STG)	0.848	0.872	0.872	**0.888**	0.882
middle temporal gyrus (MTG)	0.791	0.826	0.812	**0.838**	0.827
inferior temporal gyrus (ITG)	0.788	0.810	0.817	0.825	**0.831**
parahippocampal gyrus (PHG)	0.823	0.832	0.854	0.859	**0.865**
lingual gyrus (LG)	0.826	0.844	0.859	0.868	**0.869**
fusiform gyrus (FuG)	0.803	0.813	0.829	0.828	**0.841**
Other structures					
insular cortex (INS)	0.881	0.894	0.913	0.923	**0.927**
cingulate gyrus (CG)	0.797	0.802	0.818	0.813	**0.833**
caudate	0.900	0.914	0.919	**0.931**	0.928
putamen	0.876	0.893	0.900	0.909	**0.912**
hippocampus	0.838	0.857	0.863	**0.884**	0.877
mean	0.799	0.814	0.824	0.829	**0.837**

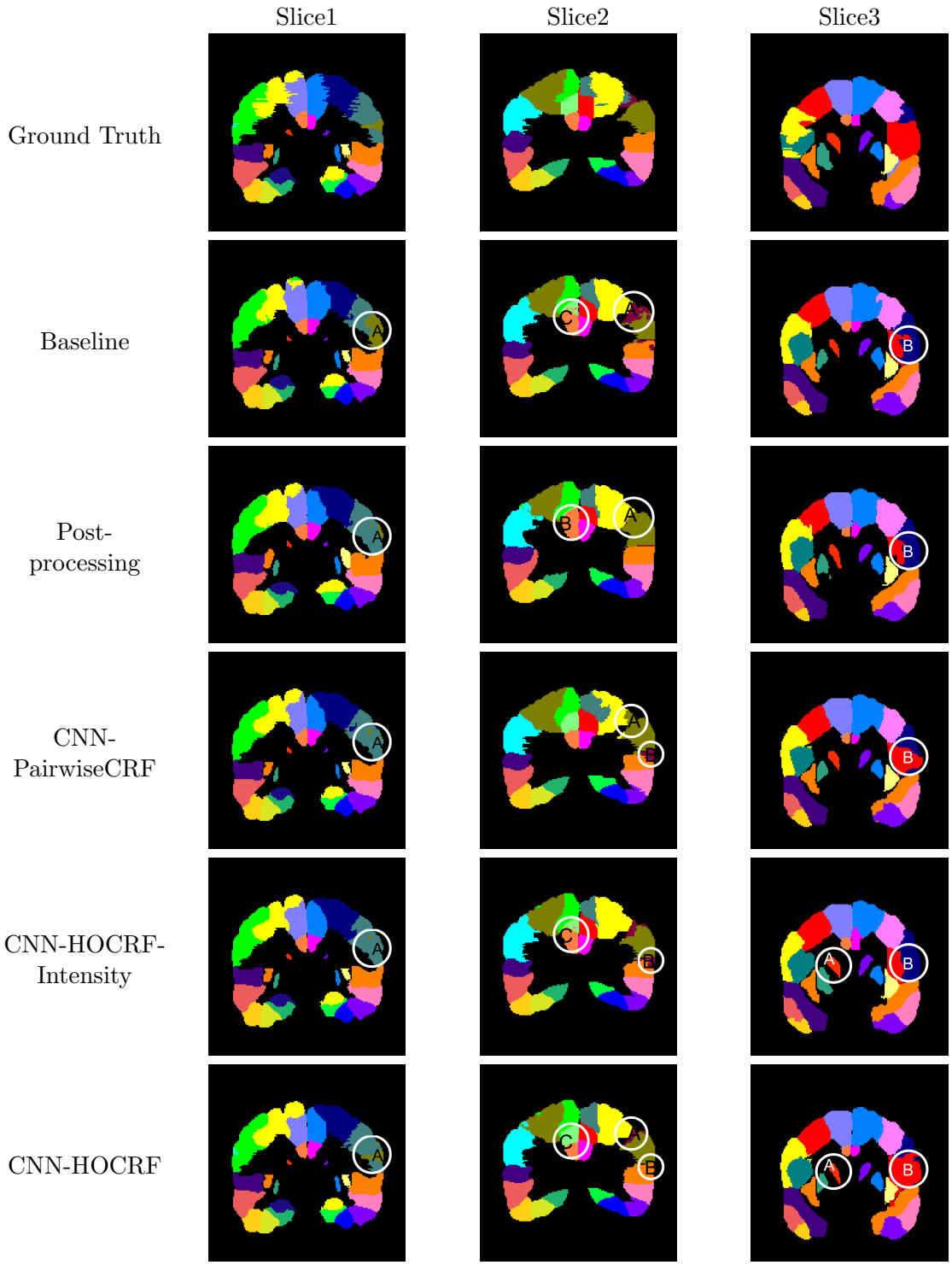


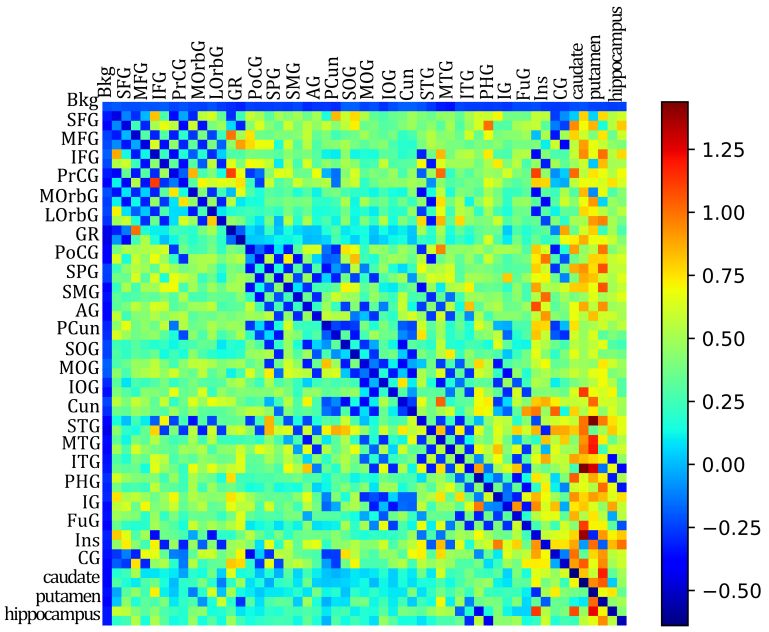
Figure 6.4: Comparision of the segmentation results of three coronal slices on LPBA40. Refer to [97] for color index.

that the pairwise clique is defined on the pixel and one of its neighborhood in $\mathcal{N}_i$, where the contextual information is implied in the pairwise potential. However, the high order clique consists of pixels that are of high probability of belonging to the same anatomical structure, which makes it difficult to learn the contextual relation from the high order term. Figure 6.5b also indicates that the importance of the high order term varies with the class. For example, for the structures with small size, e.g. putamen, hippocampus, caudate, LorbG, the high order potential are more important, resulting in a high weight.

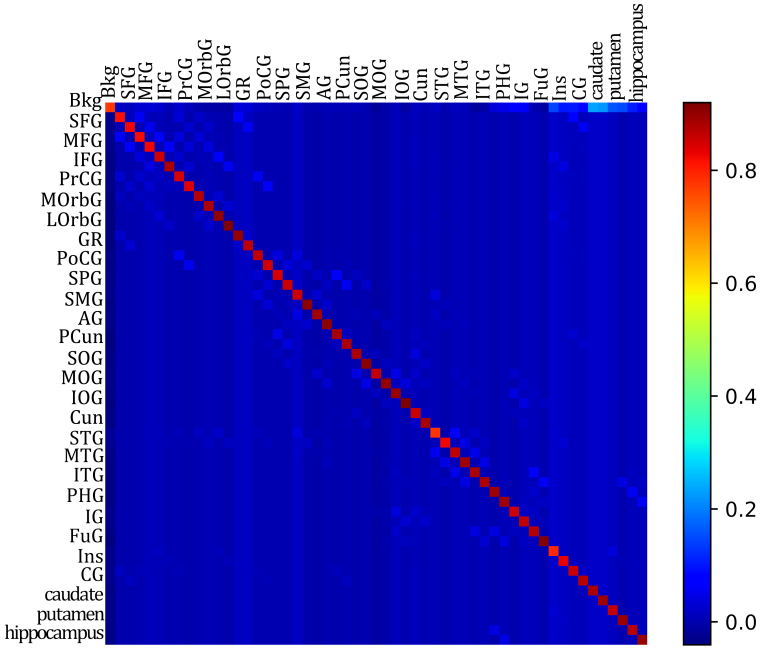
6.3.6 Integration with the State-of-the-art CNN

In this section, we combined the proposed high order CRF with the state-of-the-art CNNs. The networks for comparison include: FCN-8s, Deeplabv2 [23] and Attn-ResNet-50 in Chapter 5. For the Attn-ResNet-50, we embedded the position information in the intensity images to compensate for the loss of contextual information in the third dimension as done in Chapter 5. We set hyperparameters of the high order CRF as done in Section 6.3.4. The parameters of FCN-8s, Deeplabv2, and the encoder net of the Attn-ResNet50 are initialized with the corresponding ImageNet pre-trained models. All the CNN-HOCRF models are trained with the two-stage training strategy. We use the corresponding CNN as the baseline, which is fine-tuned for another 100 epochs after the first training stage.

The results are summarized in Table 6.4. By integrating the high order CRF (HOCRF) into the network, the segmentation performance of FCN-8s and Deeplabv2 yields significant improvements compared with the corresponding baselines, according to two-sided, paired t-tests on Dice coefficients ($p < 0.005$). However, for the Attn-ResNet-50, the improvement brought by incorporating high order CRF is not as much as the other two CNN models.



(a) label compatibility



(b) high order term weights

Figure 6.5: Learned parameters of μ and w_h. The adjacent rows (columns) in the matrix stand for the same structures of left and right hemisphere. The left and right heimisphere labels are merged for the notation of structure names in rows (columns). Refer to Table 6.3 for the full name of the anatomical structures.

Table 6.4: Mean Dice coefficient comparison on the LPBA dataset.

Networks	Dice
FCN-8s	0.799 ± 0.012
FCN-8s+HOCRF	$\mathbf{0.839 \pm 0.011}$
Deeplabv2	0.815 ± 0.012
Deeplabv2+HOCRF	$\mathbf{0.842 \pm 0.010}$
Attn-Resnet50	0.864 ± 0.010
Attn-Resnet50+HOCRF	$\mathbf{0.869 \pm 0.010}$

6.4 Conclusion

We present a novel end-to-end training CNN architecture which incorporates a high order CRF to tackle the neuroanatomical structure segmentation for brain MR images. To integrate the high order CRF into the CNN, we derive the mean field approximation update for the proposed high order CRF and model the mean field approximation as CNN layers. We also propose a label-based superpixel high order potential so that the high order cliques are adaptive during the training phase. Moreover, the semi-dense pairwise potential enables to model the mean field approximation as depthwise convolution layer.

We experimentally show that involving the high order CRF contributes to segmentation quantitatively and qualitatively. Moreover, the label based superpixel high order potential also yields superior performance over the intensity-based superpixel. By incorporating the proposed high order CRF with the modern CNN models, the performance yields significant improvement.

Chapter 7

Conclusion

This chapter presents a summary of the proposed brain anatomical segmentation methods and the conclusions inspired by them. The brain anatomical segmentation is often considered as the most core and challenging task in medical image analysis. In this dissertation, anatomical segmentation of the brain MR image has been extensively studied, and several methods, including the multi-atlas based method, graphical model, and the CNN model, have been proposed. Chapter 3 presents a two-stage majority voting scheme for hippocampus segmentation. Chapter 4 proposes a supervoxel based graphical model for whole brain segmentation. Chapter 5 develops a spatial attention model and applies the attention model to the encoder-decoder CNN architecture in order to obtain the detailed boundary segmentation results. Chapter 6 combines the strength of CNN with high order CRF and develops an end-to-end network. Finally, possible future research directions are presented in this chapter.

7.1 Contributions

The major contributions can be categorized into four groups: 1) Enhancing the traditional majority voting as a two-stage majority voting scheme. 2) Combination of the supervoxel graphical model with the multi-atlas based method for anatomical seg-

mentation of brain MR images. 3) Proposing AttentionNet which integrate spatial attention mechanism with the CNN. 4) Combination of CNN with high-order CRF for whole brain segmentation.

7.1.1 Two-stage Majority Voting

In the third chapter, we develop a novel two-stage majority voting framework for multi-atlas segmentation of hippocampus on brain MR images. The first majority voting fuses the atlas labels at the image patch level with sliding a window across the target image, followed by the second majority voting which fuses the results of the first voting for the overlapping positions. We experimentally demonstrated the effectiveness of the two-stage majority voting strategy in avoiding the over-segmentation problem compared with the original voting scheme.

7.1.2 Supervoxel Graphical Model

Multi-atlas based methods have been successfully applied to anatomical segmentation of brain MR images. However, multi-atlas based methods are sensitive to pairwise registration errors. To address this problem, we propose a supervoxel based graphical model in the fourth chapter of this dissertation, which makes the following contributions:

1. We characterize the anatomical structure segmentation of brain MR images as a supervoxel labeling through energy minimization associated with a supervoxel graphical model.

   The unary potential related to the supervoxel, i.e., the likelihood, is obtained by searching the neighborhood of the supervoxel in the atlases. Since the size of the neighborhood is defined on the supervoxels instead of voxels, the supervoxel graphical model has a large range of the searching radius compared with the patch-based technique. As a result, the proposed supervoxel graphical can bear

much more registration errors and is less sensitive to the pairwise registrations.

2. Because of the intensity overlap in MR images, we propose to perform the supervoxel segmentation on the estimated label images instead of the intensity images in order to acquire the precise supervoxel segmentation. We also experimentally show the effectiveness of the proposed supervoxel method in terms of precise aggregation.

3. The supervoxel graphical model is computationally efficient. Since supervoxels are used as nodes in the graph construction, the number of variables is much less than the graphical model defined on voxels or pixels, resulting in efficient inference.

4. We also propose a dense graphical model with high order potential to refine the results of the supervoxel graphical model.

The proposed method overcomes the challenges existing in the previous multi-atlas segmentation in terms of computational efficiency and the dependency on the complicated deformable pairwise registration. The proposed approach demonstrates superior performance over the state-of-the-art algorithms on three publicly available datasets, and significant improvement was achieved in terms of overall accuracy, per-label accuracy, and qualitative assessment.

7.1.3 AttentionNet

Encoder-decoder network is one of the most effective CNN architectures for semantic segmentation and medical image segmentation. The feature combination unit connects the high-level features from the decoder net and the finer features from the encoder net and output features that combine where and what. In Chapter 5, we propose a spatial attention model which can model the spatial dependencies between two feature maps and integrate the attention model with the encoder-decoder network to

perform anatomical segmentation on brain MR images. The proposed AttentionNet has the following contributions:

1. The spatial attention model captures the spatial dependencies between the high-level features and the finer features, selectively combines the related positions in the finer feature maps with the high-level feature maps. As a result, the attention model enables the net to highlight the relevant features to obtain a detailed prediction.

2. We develop an efficient decoder net which includes a deconvolution layer and a residual block. The decoder along with the attention model can keep the output channel to a small number. This design reduces the trainable parameters, which is efficient in alleviating the over-fitting problem for small datasets.

3. A 2D AttentionNet is developed, where the 3D relative coordinates are encoded into the intensity images. The 2D architecture benefits the net in terms of low memory requirement, deep architecture, and fine-tuning on pre-trained model. The incorporation of the spatial information not only compensates the spatial context in the third dimension but also equips the net with both intensity and spatial prior.

The proposed AttentionNet yields significant improvements in terms of segmentation accuracy and qualitative analysis compared with the other encoder-decoder networks with alternative feature combination units. Moreover, the proposed 2D AttentionNet demonstrated superior performance over the state-of-the-art 3D networks.

7.1.4 End-to-end Trainable CNN-HOCRF

Inspired by the experiment results and conclusions from Chapter 4, we add a high order potential to the pairwise CRF, resulting in a high order CRF. In order to combine the strength of CNN with the graphical model, we propose an end-to-end

trainable network which integrates CNN with a high order CRF in Chapter 6. The contributions of this work are as follows:

1. We derive the mean field approximation inference of the proposed high order CRF, which can be implemented as building blocks of CNN. Consequently, high order CRF is combined with CNN in a unified network. During the back-propagation, the derivatives of the error with respect to the parameters can be calculated and learned jointly.

2. Two modifications are made upon the energy function of high order CRF. First, we propose a semi-dense pairwise potential where the pairwise potential is defined in a pair of pixels within a neighborhood. The semi-densely pairwise potential not only agrees with the characteristic of the brain anatomical structures but also is easy to be implemented as a depthwise convolution in CNN. Second, inspired by the idea that the importance of high order potential for each class is different, we apply a class-specific weight kernel to the weight of the high order potential, increasing the flexibility of the network.

3. We develop a two-stage training strategy to train the unified network. The CNN is trained at the first training stage while the high order CRF is added for fine-tuning at the first training stage. The superpixel segments are updated during the second training stage, aiming at obtaining precise superpixel segmentation for the encouragement of the positive labeling consistency inside the sueperpxiel.

We experimentally show that involving the high order CRF contributes to the improvement of the segmentation accuracy and qualitative analysis compared with other graphical models. Moreover, the proposed high order CRF can be integrated with other modern CNN architectures and yields significant improvement.

7.2 Scope for Future Work

In this section, we thoroughly present the future works for enhancing and evaluating the proposed brain anatomical segmentation as follows:

1. Although we experimentally demonstrated the proposed 2D CNN architecture achieves significant improvement over the 3D architecture. However, 3D networks are capable of capturing 3D contextual information, which plays an important role in MR image segmentation. In the future, based on the proposed AttentionNet, we plan to develop the 3D model for anatomical brain segmentation.

2. Currently, all the datasets involved in this thesis are small, which hinders the performance of the network. We target to develop a transfer learning strategy which can transfer the knowledge among datasets with different labeling protocols. For example, MICCAI2012 and LPBA have different labeling protocols but similar intensity images. We aim to develop a transfer learning strategy which can deal with training data with varying protocols of labeling, in order to enlarge the size of the training data.

3. In brain segmentation field, the tumor segmentation is another important research field. The brain tumor segmentation is of larger variation in terms of the shape, appearance, and volume compared with the anatomical segmentation problem. We plan to modify and apply the AttentionNet to address the tumor segmentation in brain MR images.

Bibliography

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

[2] Zeynettin Akkus, Alfiia Galimzianova, Assaf Hoogi, Daniel L Rubin, and Bradley J Erickson. Deep learning for brain mri segmentation: state of the art and future directions. *Journal of digital imaging*, 30(4):449–459, 2017.

[3] Stavros Alchatzidis, Aristeidis Sotiras, Evangelia I Zacharaki, and Nikos Paragios. A discrete mrf framework for integrated multi-atlas registration and segmentation. *International Journal of Computer Vision*, 121(1):169–181, 2017.

[4] Paul Aljabar, R Heckemann, Alexander Hammers, Joseph V Hajnal, and Daniel Rueckert. Classifier selection strategies for label fusion using large atlas databases. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 523–531. Springer, 2007.

[5] Bjoern Andres, Ullrich Koethe, Thorben Kroeger, Moritz Helmstaedter, Kevin L Briggman, Winfried Denk, and Fred A Hamprecht. 3d segmentation of sbfsem images of neuropil by a graphical model over supervoxel boundaries. *Medical image analysis*, 16(4):796–805, 2012.

[6] Xabier Artaechevarria, Arrate Muñoz-Barrutia, and Carlos Ortiz-de Solórzano. Efficient classifier generation and weighted voting for atlas-based segmentation: Two small steps faster and closer to the combination oracle. In *Medical Imaging*

2008: Image Processing, volume 6914, page 69141W. International Society for Optics and Photonics, 2008.

[7] Xabier Artaechevarria, Arrate Munoz-Barrutia, and Carlos Ortiz-de Solórzano. Combination strategies in multi-atlas image segmentation: application to brain mr data. *IEEE transactions on medical imaging*, 28(8):1266–1277, 2009.

[8] Andrew J Asman and Bennett A Landman. Non-local statistical label fusion for multi-atlas segmentation. *Medical image analysis*, 17(2):194–208, 2013.

[9] Andrew J Asman and Bennett A Landman. Hierarchical performance estimation in the statistical label fusion framework. *Medical image analysis*, 18(7):1070–1081, 2014.

[10] Brian B Avants, Charles L Epstein, Murray Grossman, and James C Gee. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis*, 12(1):26–41, 2008.

[11] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[12] Wenjia Bai, Wenzhe Shi, Christian Ledig, and Daniel Rueckert. Multi-atlas segmentation with augmented features for cardiac mr images. *Medical image analysis*, 19(1):98–109, 2015.

[13] Wenjia Bai, Wenzhe Shi, Declan P O'Regan, Tong Tong, Haiyan Wang, Shahnaz Jamil-Copley, Nicholas S Peters, and Daniel Rueckert. A probabilistic patch-based label fusion model for multi-atlas segmentation with registration refinement: application to cardiac mr images. *IEEE transactions on medical imaging*, 32(7):1302–1315, 2013.

[14] Siqi Bao and Albert CS Chung. Multi-scale structured cnn with label consistency for brain mr image segmentation. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 6(1):113–117, 2018.

[15] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61(2):139–157, 2005.

[16] Boubakeur Belaroussi, Julien Milles, Sabin Carme, Yue Min Zhu, and Hugues Benoit-Cattin. Intensity non-uniformity correction in mri: existing methods and their validation. *Medical image analysis*, 10(2):234–246, 2006.

[17] Peter A Brex, Olga Ciccarelli, Jonathon I O'Riordan, Michael Sailer, Alan J Thompson, and David H Miller. A longitudinal study of abnormalities on mri and disability from multiple sclerosis. *New England Journal of Medicine*, 346(3):158–164, 2002.

[18] Yihui Cao, Yuan Yuan, Xuelong Li, and Pingkun Yan. Putting images on a manifold for atlas-based image segmentation. In *2011 18th IEEE International Conference on Image Processing*, pages 289–292. IEEE, 2011.

[19] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[20] Hao Chen, Qi Dou, Lequan Yu, Jing Qin, and Pheng-Ann Heng. Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images. *NeuroImage*, 170:446–455, 2018.

[21] Hao Chen, Xiao Juan Qi, Jie Zhi Cheng, and Pheng Ann Heng. Deep contextual networks for neuronal structure segmentation. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

[22] Hao Chen, Xiaojuan Qi, Lequan Yu, and Pheng-Ann Heng. Dcan: deep contour-aware networks for accurate gland segmentation. In *Proceedings of the IEEE*

conference on Computer Vision and Pattern Recognition, pages 2487–2496, 2016.

[23] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.

[24] Dawn C Collier, Stuart SC Burnett, Mayankkumar Amin, Stephen Bilton, Christopher Brooks, Amanda Ryan, Dominique Roniger, Danny Tran, and George Starkschall. Assessment of consistency in contouring of normal-tissue anatomic structures. *Journal of applied clinical medical physics*, 4(1):17–24, 2003.

[25] Pierrick Coupé, José V Manjón, Vladimir Fonov, Jens Pruessner, Montserrat Robles, and D Louis Collins. Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. *NeuroImage*, 54(2):940–954, 2011.

[26] Pierrick Coupé, Pierre Yger, Sylvain Prima, Pierre Hellier, Charles Kervrann, and Christian Barillot. An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images. *IEEE transactions on medical imaging*, 27(4):425–441, 2008.

[27] Shusil Dangi, Ziv Yaniv, and Cristian Linte. A distance map regularized cnn for cardiac cine mr image segmentation. *arXiv preprint arXiv:1901.01238*, 2019.

[28] Alexander de Brebisson and Giovanni Montana. Deep neural networks for anatomical brain segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–28, 2015.

[29] Ivana Despotović, Bart Goossens, and Wilfried Philips. Mri segmentation of the human brain: challenges, methods, and applications. *Computational and mathematical methods in medicine*, 2015, 2015.

[30] Bradford C Dickerson, I Goncharova, MP Sullivan, C Forchetti, RS Wilson, DA Bennett, Laurel A Beckett, et al. Mri-derived entorhinal and hippocampal atrophy in incipient and very mild alzheimers disease. *Neurobiology of aging*, 22(5):747–754, 2001.

[31] Jose Dolz, Ismail Ben Ayed, Jing Yuan, and Christian Desrosiers. Isointense infant brain segmentation with a hyper-dense connected convolutional neural network. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 616–620. IEEE, 2018.

[32] Jose Dolz, Christian Desrosiers, and Ismail Ben Ayed. 3d fully convolutional networks for subcortical segmentation in mri: A large-scale study. *NeuroImage*, 170:456–470, 2018.

[33] Michal Drozdzal, Gabriel Chartrand, Eugene Vorontsov, Mahsa Shakeri, Lisa Di Jorio, An Tang, Adriana Romero, Yoshua Bengio, Chris Pal, and Samuel Kadoury. Learning normalized inputs for iterative estimation in medical image segmentation. *Medical image analysis*, 44:1–13, 2018.

[34] Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

[35] Bruce Fischl, David H Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Van Der Kouwe, Ron Killiany, David Kennedy, Shuna Klaveness, et al. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002.

[36] Tobias Gass, Gabor Szekely, and Orcun Goksel. Multi-atlas segmentation and landmark localization in images with large field of view. In *International MICCAI Workshop on Medical Computer Vision*, pages 171–180. Springer, 2014.

[37] Tobias Gass, Gabor Szekely, and Orcun Goksel. Simultaneous segmentation and multiresolution nonrigid atlas registration. *IEEE Transactions on Image Processing*, 23(7):2931–2943, 2014.

[38] Patric Hagmann, Leila Cammoun, Xavier Gigandet, Reto Meuli, Christopher J Honey, Van J Wedeen, and Olaf Sporns. Mapping the structural core of human cerebral cortex. *PLoS biology*, 6(7):e159, 2008.

[39] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35:18 – 31, 2017.

[40] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[41] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[42] Rolf A Heckemann, Joseph V Hajnal, Paul Aljabar, Daniel Rueckert, and Alexander Hammers. Automatic anatomical brain mri segmentation combining label propagation and decision fusion. *NeuroImage*, 33(1):115–126, 2006.

[43] Mattias P Heinrich, Oskar Maier, and Heinz Handels. Multi-modal multi-atlas segmentation using discrete optimisation and self-similarities. In *VISCERAL Challenge@ ISBI*, pages 27–30, 2015.

[44] Mattias P Heinrich, Ivor JA Simpson, BartŁomiej W Papież, Michael Brady, and Julia A Schnabel. Deformable image registration by combining uncertainty estimates from supervoxel belief propagation. *Medical image analysis*, 27:57–71, 2016.

[45] Derek LG Hill, Philipp G Batchelor, Mark Holden, and David J Hawkes. Medical image registration. *Physics in medicine & biology*, 46(3):R1, 2001.

[46] Zujun Hou. A review on mr image intensity inhomogeneity correction. *International journal of biomedical imaging*, 2006, 2006.

[47] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[48] Jie Huo, Guanghui Wang, QM Jonathan Wu, and Akilan Thangarajah. Label fusion for multi-atlas segmentation based on majority voting. In *International Conference Image Analysis and Recognition*, pages 100–106. Springer, 2015.

[49] Jie Huo, Jonathan Wu, Jiuwen Cao, and Guanghui Wang. Supervoxel based method for multi-atlas segmentation of brain mr images. *NeuroImage*, 175:201–214, 2018.

[50] Juan Eugenio Iglesias and Mert R Sabuncu. Multi-atlas segmentation of biomedical images: a survey. *Medical image analysis*, 24(1):205–219, 2015.

[51] Juan Eugenio Iglesias, Mert Rory Sabuncu, and Koen Van Leemput. A probabilistic, non-parametric framework for inter-modality label fusion. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 576–583. Springer, 2013.

[52] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[53] Benjamin Irving, James M Franklin, Bartłomiej W Papież, Ewan M Anderson, Ricky A Sharma, Fergus V Gleeson, Michael Brady, and Julia A Schnabel. Pieces-of-parts for supervoxel segmentation with global context: Application to dce-mri tumour delineation. *Medical image analysis*, 32:69–83, 2016.

[54] Aurélie Isambert, Frédéric Dhermain, François Bidault, Olivier Commowick, Pierre-Yves Bondiau, Grégoire Malandain, and Dimitri Lefkopoulos. Evaluation

of an atlas-based automatic segmentation software for the delineation of brain organs at risk in a radiation therapy clinical context. *Radiotherapy and oncology*, 87(1):93–99, 2008.

[55] Ivana Isgum, Marius Staring, Annemarieke Rutten, Mathias Prokop, Max A Viergever, and Bram Van Ginneken. Multi-atlas-based segmentation with local decision fusionapplication to cardiac and aortic segmentation in ct scans. *IEEE transactions on medical imaging*, 28(7):1000–1010, 2009.

[56] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194, 2001.

[57] Clifford R Jack Jr, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L. Whitwell, Chadwick Ward, et al. The alzheimer's disease neuroimaging initiative (adni): Mri methods. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 27(4):685–691, 2008.

[58] Hongjun Jia, Pew-Thian Yap, and Dinggang Shen. Iterative multi-atlas-based multi-image segmentation with tree-based registration. *NeuroImage*, 59(1):422–430, 2012.

[59] Yacine Kabir, Michel Dojat, Benoît Scherrer, Florence Forbes, and Catherine Garbay. Multimodal mri segmentation of ischemic stroke lesions. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1595–1598. IEEE, 2007.

[60] Samuel Kadoury, Nadine Abi-Jaoudeh, and Pablo A Valdes. Higher-order crf tumor segmentation with discriminant manifold potentials. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 719–726. Springer, 2013.

[61] Konstantinos Kamnitsas, Christian Ledig, Virginia FJ Newcombe, Joanna P Simpson, Andrew D Kane, David K Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical image analysis*, 36:61–78, 2017.

[62] Vladimir Katkovnik, Alessandro Foi, Karen Egiazarian, and Jaakko Astola. From local kernel to nonlocal multiple-model image denoising. *International journal of computer vision*, 86(1):1, 2010.

[63] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[64] Lisa M Koch, Martin Rajchl, Wenjia Bai, Christian F Baumgartner, Tong Tong, Jonathan Passerat-Palmbach, Paul Aljabar, and Daniel Rueckert. Multi-atlas segmentation using partially annotated data: Methods and annotation strategies. *arXiv preprint arXiv:1605.00029*, 2016.

[65] Pushmeet Kohli, Philip HS Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

[66] Daphne Koller, Nir Friedman, and Francis Bach. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[67] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE transactions on pattern analysis and machine intelligence*, 28(10):1568–1583, 2006.

[68] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[69] B Landman and S Warfield. Miccai 2012 workshop on multi-atlas labeling. In *Medical image computing and computer assisted intervention conference*, 2012.

[70] Bennett A Landman, Andrew J Asman, Andrew G Scoggins, John A Bogovic, Fangxu Xing, and Jerry L Prince. Robust statistical fusion of image labels. *IEEE transactions on medical imaging*, 31(2):512–522, 2012.

[71] Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1):29–44, 2001.

[72] Guosheng Lin, Chunhua Shen, Anton Van Den Hengel, and Ian Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3194–3203, 2016.

[73] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen Change Loy, and Xiaoou Tang. Deep learning markov random field for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1814–1828, 2018.

[74] Xavier Lladó, Arnau Oliver, Mariano Cabezas, Jordi Freixenet, Joan C Vilanova, Ana Quiles, Laia Valls, Lluís Ramió-Torrentà, and Àlex Rovira. Segmentation of multiple sclerosis lesions in brain mri: a review of automated approaches. *Information Sciences*, 186(1):164–185, 2012.

[75] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[76] Jyrki MP Lötjönen, Robin Wolz, Juha R Koikkalainen, Lennart Thurfjell, Gunhild Waldemar, Hilkka Soininen, Daniel Rueckert, Alzheimer's Disease Neuroimaging Initiative, et al. Fast and robust multi-atlas segmentation of brain magnetic resonance images. *Neuroimage*, 49(3):2352–2365, 2010.

[77] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on mri. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019.

[78] Oskar Maier, Bjoern H Menze, Janina von der Gablentz, Levin Häni, Mattias P Heinrich, Matthias Liebrand, Stefan Winzeck, Abdul Basit, Paul Bentley, Liang Chen, et al. Isles 2015-a public evaluation benchmark for ischemic stroke lesion segmentation from multispectral mri. *Medical image analysis*, 35:250–269, 2017.

[79] Pablo Márquez-Neila, Pushmeet Kohli, Carsten Rother, and Luis Baumela. Non-parametric higher-order random fields for image segmentation. In *European Conference on Computer Vision*, pages 269–284. Springer, 2014.

[80] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

[81] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014.

[82] Pim Moeskops, Max A Viergever, Adriënne M Mendrik, Linda S de Vries, Manon JNL Benders, and Ivana Išgum. Automatic segmentation of mr brain images with a convolutional neural network. *IEEE transactions on medical imaging*, 35(5):1252–1261, 2016.

[83] Sérgio Pereira, Adriano Pinto, Victor Alves, and Carlos A Silva. Brain tumor segmentation using convolutional neural networks in mri images. *IEEE transactions on medical imaging*, 35(5):1240–1251, 2016.

[84] Martin Rajchl, John SH Baxter, A Jonathan McLeod, Jing Yuan, Wu Qiu, Terry M Peters, and Ali R Khan. Hierarchical max-flow segmentation framework for multi-atlas segmentation with kohonen self-organizing map based gaussian mixture modeling. *Medical image analysis*, 27:45–56, 2016.

[85] Liliane Ramus, Olivier Commowick, and Grégoire Malandain. Construction of patient specific atlases from locally most similar anatomical pieces. In *In-*

ternational Conference on Medical Image Computing and Computer-Assisted Intervention, pages 155–162. Springer, 2010.

[86] Torsten Rohlfing, Robert Brandt, Randolf Menzel, and Calvin R Maurer Jr. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. *NeuroImage*, 21(4):1428–1442, 2004.

[87] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[88] Françcois Rousseau, Piotr A Habas, and Colin Studholme. A supervised patch-based approach for human brain labeling. *IEEE transactions on medical imaging*, 30(10):1852–1862, 2011.

[89] Àlex Rovira and Adelaida León. Mr in the diagnosis and monitoring of multiple sclerosis: an overview. *European journal of radiology*, 67(3):409–414, 2008.

[90] Chris Russell, Pushmeet Kohli, Philip HS Torr, et al. Associative hierarchical crfs for object class image segmentation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 739–746. IEEE, 2009.

[91] Mert R Sabuncu, BT Thomas Yeo, Koen Van Leemput, Bruce Fischl, and Polina Golland. A generative model for image segmentation based on label fusion. *IEEE transactions on medical imaging*, 29(10):1714–1729, 2010.

[92] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Auto-context convolutional neural network (auto-net) for brain extraction in magnetic resonance imaging. *IEEE Transactions on Medical Imaging*, 2017.

[93] Benoit Scherrer, Florence Forbes, Catherine Garbay, and Michel Dojat. Fully bayesian joint model for mr brain scan tissue and structure segmentation. In

International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 1066–1074. Springer, 2008.

[94] Mohak Shah, Yiming Xiao, Nagesh Subbanna, Simon Francis, Douglas L Arnold, D Louis Collins, and Tal Arbel. Evaluating intensity normalization on mris of human brain with multiple sclerosis. *Medical image analysis*, 15(2):267–282, 2011.

[95] Mohak Shah, Yiming Xiao, Nagesh Subbanna, Simon Francis, Douglas L Arnold, D Louis Collins, and Tal Arbel. Evaluating intensity normalization on mris of human brain with multiple sclerosis. *Medical image analysis*, 15(2):267–282, 2011.

[96] Zu Y Shan, Guang H Yue, and Jing Z Liu. Automated histogram-based brain segmentation in t1-weighted three-dimensional magnetic resonance head images. *NeuroImage*, 17(3):1587–1598, 2002.

[97] David W Shattuck, Mubeena Mirza, Vitria Adisetiyo, Cornelius Hojatkashani, Georges Salamon, Katherine L Narr, Russell A Poldrack, Robert M Bilder, and Arthur W Toga. Construction of a 3d probabilistic atlas of human cortical structures. *Neuroimage*, 39(3):1064–1080, 2008.

[98] Feng Shi, Pew-Thian Yap, Yong Fan, John H Gilmore, Weili Lin, and Dinggang Shen. Construction of multi-region-multi-reference atlases for neonatal brain mri segmentation. *Neuroimage*, 51(2):684–693, 2010.

[99] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision*, 81(1):2–23, 2009.

[100] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Learning local feature descriptors using convex optimisation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1573–1585, 2014.

[101] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[102] Stephen M Smith, Mark Jenkinson, Mark W Woolrich, Christian F Beckmann, Timothy EJ Behrens, Heidi Johansen-Berg, Peter R Bannister, Marilena De Luca, Ivana Drobnjak, David E Flitney, et al. Advances in functional and structural mr image analysis and implementation as fsl. *Neuroimage*, 23:S208–S219, 2004.

[103] Zhiqiang Tian, Lizhi Liu, and Baowei Fei. Deep convolutional neural network for prostate mr segmentation. In *Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10135, page 101351L. International Society for Optics and Photonics, 2017.

[104] Joseph Tighe and Svetlana Lazebnik. Superparsing: scalable nonparametric image parsing with superpixels. In *European conference on computer vision*, pages 352–365. Springer, 2010.

[105] Tong Tong, Robin Wolz, Pierrick Coupé, Joseph V Hajnal, Daniel Rueckert, Alzheimer's Disease Neuroimaging Initiative, et al. Segmentation of mr images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. *NeuroImage*, 76:11–23, 2013.

[106] Nicholas J Tustison, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich, and James C Gee. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310–1320, 2010.

[107] Fedde van der Lijn, Tom den Heijer, Monique MB Breteler, and Wiro J Niessen. Hippocampus segmentation in mr images using atlas registration, voxel classification, and graph cuts. *Neuroimage*, 43(4):708–720, 2008.

[108] Koen Van Leemput, Frederik Maes, Dirk Vandermeulen, and Paul Suetens. Automated model-based tissue classification of mr images of the brain. *IEEE transactions on medical imaging*, 18(10):897–908, 1999.

[109] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[110] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Diffeomorphic demons: Efficient non-parametric image registration. *NeuroImage*, 45(1):S61–S72, 2009.

[111] Vibhav Vineet, Jonathan Warrell, and Philip HS Torr. Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. *International Journal of Computer Vision*, 110(3):290–307, 2014.

[112] Christian Wachinger, Martin Reuter, and Tassilo Klein. Deepnat: Deep convolutional neural network for segmenting neuroanatomy. *NeuroImage*, 170:434–445, 2017.

[113] Christian Wachinger, Martin Reuter, and Tassilo Klein. Deepnat: deep convolutional neural network for segmenting neuroanatomy. *NeuroImage*, 170:434–445, 2018.

[114] Chaohui Wang, Olivier Teboul, Fabrice Michel, Salma Essafi, and Nikos Paragios. 3d knowledge-based segmentation using pose-invariant higher-order graphs. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 189–196. Springer, 2010.

[115] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2017.

[116] Hongzhi Wang, Yu Cao, and Tanveer Syeda-Mahmood. Multi-atlas segmentation with learning-based label fusion. In *International Workshop on Machine Learning in Medical Imaging*, pages 256–263. Springer, 2014.

[117] Hongzhi Wang, Jung W Suh, Sandhitsu R Das, John B Pluta, Caryne Craige, and Paul A Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE transactions on pattern analysis and machine intelligence*, 35(3):611–623, 2013.

[118] Hongzhi Wang and Paul Yushkevich. Multi-atlas segmentation with joint label fusion and corrective learningan open source implementation. *Frontiers in neuroinformatics*, 7:27, 2013.

[119] Hongzhi Wang and Paul A Yushkevich. Multi-atlas segmentation without registration: a supervoxel-based approach. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 535–542. Springer, 2013.

[120] Simon K Warfield, Kelly H Zou, and William M Wells. Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, 23(7):903, 2004.

[121] Robin Wolz, Chengwen Chu, Kazunari Misawa, Michitaka Fujiwara, Kensaku Mori, and Daniel Rueckert. Automated abdominal multi-organ segmentation with subject-specific atlas generation. *IEEE transactions on medical imaging*, 32(9):1723–1730, 2013.

[122] Guorong Wu, Minjeong Kim, Gerard Sanroma, Qian Wang, Brent C Munsell, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, et al. Hierarchical multi-atlas label fusion with multi-scale feature representation and label-specific patch partition. *NeuroImage*, 106:34–46, 2015.

[123] Guorong Wu, Qian Wang, Daoqiang Zhang, Feiping Nie, Heng Huang, and Dinggang Shen. A generative probability model of joint label fusion for multi-atlas based brain segmentation. *Medical image analysis*, 18(6):881–890, 2014.

[124] Minjie Wu, Caterina Rosano, Pilar Lopez-Garcia, Cameron S Carter, and Howard J Aizenstein. Optimum template selection for atlas-based segmentation. *NeuroImage*, 34(4):1612–1618, 2007.

[125] Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11(Oct):2543–2596, 2010.

[126] Lequan Yu, Jie-Zhi Cheng, Qi Dou, Xin Yang, Hao Chen, Jing Qin, and Pheng-Ann Heng. Automatic 3d cardiovascular mr segmentation with densely-connected volumetric convnets. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 287–295. Springer, 2017.

[127] Lequan Yu, Xin Yang, Hao Chen, Jing Qin, and Pheng Ann Heng. Volumetric convnets with mixed residual connections for automated prostate segmentation from 3d mr images. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

[128] Ning Yu, Hongzhi Wang, and Paul A Yushkevich. Supervoxel-based hierarchical markov random field framework for multi-atlas segmentation. In *International Workshop on Patch-based Techniques in Medical Imaging*, pages 100–108. Springer, 2016.

[129] Wenlu Zhang, Rongjian Li, Houtao Deng, Li Wang, Weili Lin, Shuiwang Ji, and Dinggang Shen. Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage*, 108:214–224, 2015.

[130] Shuai Zheng, Ming-Ming Cheng, Jonathan Warrell, Paul Sturgess, Vibhav Vineet, Carsten Rother, and Philip HS Torr. Dense semantic image segmentation with objects and attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3214–3221, 2014.

[131] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.

[132] Darko Zikic, Ben Glocker, and Antonio Criminisi. Encoding atlases by ran-

domized classification forests for efficient multi-atlas label propagation. *Medical image analysis*, 18(8):1262–1273, 2014.

Appendix A

Springer Permission to Reprint

In reference to Springer copyrighted material which is used with permission in this thesis, the Springer does not endorse any of University of Windsor's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing Springer copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to `https://www.springer.com/gp/rights-permissions/obtaining-permissions/882` to learn how to obtain a License from RightsLink.

Appendix B

Elsevier Permission to Reprint

In reference to Elsevier copyrighted material which is used with permission in this thesis, the Elsevier does not endorse any of University of Windsors products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing Elsevier copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please visit: `https://www.elsevier.com/about/our-business/policies/copyright#Author-rights`.

Vita Auctoris

Name : Jie Huo

Education :

2019 Doctor of Philosophy
Electrical and Computer Engineering
University of Windsor, Canada

2014 Master of Engineering (Thesis)
Biomedical Engeneering
Tianjin University, China

2011 Bachelor of Engineering
Biomedical Engeneering
Tianjin University, China