Downloaded from orbit.dtu.dk on: May 06, 2024

DTU Library

# Computational Analysis of Brain Images: Towards a Useful Tool in Clinical Practice

**Puonti, Oula**

*Publication date:*
2016

*Document Version*
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

*Citation (APA):*
Puonti, O. (2016). *Computational Analysis of Brain Images: Towards a Useful Tool in Clinical Practice*. Technical University of Denmark. DTU Compute PHD-2015 No. 396

**General rights**
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Computational Analysis of Brain Images: Towards a Useful Tool in Clinical Practice

Oula Puonti


Kongens Lyngby 2015
PhD-2015-396

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Matematiktorvet, building 303B,
2800 Kongens Lyngby, Denmark
Phone +45 4525 3351
compute@compute.dtu.dk
www.compute.dtu.dk PhD-2015-396

# Summary (English)

Due to its excellent soft tissue contrast and versatility, magnetic resonance imaging (MRI) has become arguably the most important tool for studying the structure and disorders of the human brain. Although in recent years tremendous advances have been made in automatic segmentation of brain MRI scans, many of the developed methods are not readily extendible to clinical applications due to the variability of clinical MRI data and the presence of pathologies, such as tumors or lesions. Thus, clinicians are forced to manually analyze the MRI data, which is a time consuming task and introduces rater-dependent variability that reduces the accuracy and sensitivity of the results.

The goal of this PhD-project was to enlarge the scope of the automatic tools into clinical applications. In order to tackle the variability of the data and presence of pathologies, we base our methods on Bayesian generative modeling, which combines detailed prior models of the human neuroanatomy and pathologies with models of the MRI imaging process. This approach allows us to describe the observed MRI data in a principled manner, and to integrate explicit models of different disease effects and imaging artifacts into the framework when needed.

This thesis presents an introduction to the theory behind the generative modeling approach, and an overview of the main results. The first part concentrates on segmenting different neuroanatomical structures in MRI scans of healthy subjects, and the second part describes how this framework can be extended with models of brain lesions. This results in a set of fast, robust and fully automatic tools for segmenting MRI brain scans of both healthy subjects and subjects suffering from brain disorders such as multiple sclerosis. Having access to quantitative measures of both lesions and the surrounding structures opens

up avenues for clinicians to study the effect of these type of disorders on the full brain anatomy. This could potentially help in discovering sensitive biomarkers for early diagnosis and tracking of disease development.

# Summary (Danish)

Grundet dets fremragende kontrast i blødt væv, er magnetic resonance imaging (MRI) blevet den dominante billedmodalitet til at studere struktur samt patologi i den menneskelige hjerne. Selvom de seneste år har set betragtelige fremskridt inden for automatisk segmentering af MRI skanninger af hjernen, er mange af de udviklede metoder endnu ikke klar til klinisk brug grundet variationen i MRI data samt tilstedeværelsen af patologier, såsom tumorer og læsioner. Derfor er klinikerne tvunget til at analysere MRI dataen manuelt, hvilket er en tidskrævende opgave. Samtidigt introducerer dette variabilitet i analysen som afhænger af klinikeren. Dette påvirker nøjagtighed samt sensitivitet i resultaterne.

Målet med dette PhD projekt var at udvide anvendelsen af de automatiserede værktøjer til klinisk brug. For at håndtere variabiliteten i data samt tilstedeværelsen af patologi, baserer vi vores metoder på et framework bestående af Bayesiansk, generativ modellering. Dette framework kombinerer detaljerede a-priori modeller af den menneskelige neuro-anatomi samt patologier, med modeller af MRI billeddannelses-processen. Ved at følge denne tilgang kan den observerede MRI data beskrives på formaliseret vis, og modeller af forskellige patologiske effekter samt billedartefakter kan integreres eksplicit i frameworket hvis nødvendigt.

Afhandlingen præsenterer en introduktion til teorien bag den anvendte modellering, samt et overblik over de primære bidrag i projektet. Den første del fokuserer på segmentering af de forskellige neuro-anatomiske strukturer i MRI skanninger af raske individer, og den anden del beskriver hvordan dette framework kan udvides med modeller af læsioner. Dette resulterer i en gruppe af hurtige, stabile og fuldt automatiserede værktøjer til segmentering af MRI scans af den men-

neskelige hjerne i både raske individer samt patienter der lider af sygdomme såsom multipel sklerose. Med adgang til kvantitative mål af både læsioner og omgivende strukturer tillades klinikere at studere effekten af denne type patologier i den totale neuro-anatomi. Dette kan potentielt hjælpe til at opdage sensitive biomarkører til tidlig diagnosticering samt overvågning af progression af patologier.

# Preface

This thesis was prepared at the department of Applied Mathematics and Computer Science at the Technical University of Denmark in partial fulfilment of the requirements for acquiring a PhD in Applied Mathematics with an emphasis on Image Analysis. The thesis was prepared with funding solely from the Technical University of Denmark with associate professor Koen Van Leemput as main supervisor and professor Rasmus Larsen as co-supervisor.

The thesis deals with automated segmentation methods for clinically acquired magnetic resonance images of the human brain.

Lyngby, 31-October-2015

Oula Puonti

# Acknowledgements

Firstly, I would like to express my gratitude to my supervisor Koen Van Leemput for the continuous support during my PhD studies, for his patience, motivation, and vast knowledge in the field of medical imaging. I could not have imagined having a better advisor and mentor for my PhD studies.

My sincere thanks also goes to my co-supervisor professor Rasmus Larsen and to professor Erkki Oja. The latter for hosting me in his lab for three months during my external stay.

I would also like to thank my fellow lab mates and colleagues for creating a pleasant and motivating working environment, and for many good times. A very special thanks goes to Christian Thode Larsen, Mikael Agn, Daniel Andreasen, Anders Nymark Christensen, Jakob Schack Vestergaard, Mark Lyksborg and Anders Boesen Lindbo Larsen, all of whom have helped me in one way or the other during the past three years. Many inspiring discussions were held and their inputs have been greatly appreciated. Furthermore, I would like to thank Eugenio Iglesias for continuous collaboration and support in problems big and small.

Above all I would like to thank my family and girlfriend for supporting me, not only for the past three years, but thus far in my journey through life.

# Scientific contributions

## Papers included in this thesis

**Paper A** O. Puonti, J. E. Iglesias and K. Van Leemput. Fast, Sequence Adaptive Parcellation of Brain MR Using Parametric Models. In *Medical Image Computing and Computer Assisted Intervention – MICCAI – 16th International Conference, Nagoya, Japan, September 22-26, 2013, Proceedings, Part I. Springer, pages 727 – 734, Lecture Notes in Computer Science, Vol. 8149.*

**Paper B** O. Puonti, J. E. Iglesias and K. Van Leemput. Fast and Sequence-Adaptive Whole-Brain Segmentation Using Parametric Bayesian Modeling. *To be submitted to NeuroImage.*

**Paper C** O. Puonti and K. Van Leemput. Simultaneous Whole-Brain Segmentation and White Matter Lesion Detection Using Contrast-Adaptive Probabilistic Models. *To appear in Proceedings of the Brain Lesions Workshop, 2015, Lecture Notes in Computer Science.*

This is an extended version of the peer-reviewed article presented in the BrainLes workshop at the 18th International Conference on Medical Image Computing and Computer Assisted Intervention – MICCAI.

**Paper D** M. Agn, O. Puonti, P. Munck af Rosenschöld, I. Law and K. Van Leemput. Brain Tumor Segmentation by a Generative Model with a Prior on Tumor Shape. *An extended version to appear in Proceedings of the Brain Lesions Workshop, 2015, Lecture Notes in Computer Science.*

This work was presented in the Multimodal Brain Tumor Segmentation Challenge, where it was ranked the 3rd overall best tumor segmentation

method while being the best among fully automated methods. We were invited to submit an extended version of the article which will appear in the post-proceedings of the challenge.

**Paper E** M. Lyksborg, O. Puonti, M. Agn and R. Larsen. An Ensemble of 2D Convolutional Neural Networks for Tumor Segmentation. *Proceedings of the 19th Scandinavian Conference on Image Analysis, SCIA 2015 (ISBN: 978-3-319-19664-0), pages: 201-211, 2015, Springer, Lecture Notes in Computer Science, Vol. 9127.*

**Paper F** K. Van Leemput and O. Puonti. Tissue Classification. *Brain Mapping: An Encyclopedic Reference. A. C. Toga, ed., 2015, Elsevier.*

# Contents

Chapter 1

# Motivation

Since the first magnetic resonance (MR) scanning of the human head was conducted in 1978 by Clow and Young [Gev06], the use of MR imaging (MRI) for studying the structure of the human brain has grown exponentially in clinics and research centers all around the world. The main attraction with MRI is its in-built ability to show the fine anatomical structures of the brain in exquisite detail, as well as its excellent soft tissue contrast. Furthermore, the various different scan sequences developed for MRI help to highlight many different biological properties of the brain tissue being imaged. This is especially important in hospitals and clinics, where the main interest is in studying and diagnosing the disorders of the brain. The complementary information provided by the different scan-sequences is essential for assessing the full extent of the brain damage in pathologies such as tumors. This is visualized in figure 1.1, where each individual scan-sequence shows different compartments of the tumor. Thus in routine clinical practice, a typical scan session consists of obtaining multiple MR images with different scan-sequences yielding so-called multi-contrast scans, i.e., a multitude of three-dimensional images with different contrast-properties.

In recent years, tremendous advances have been made in automatically segmenting the type of anatomical brain MR scans that are used in neuroscientific studies. However, the development of these automated methods has been mainly aimed at segmenting a specific type of MR contrast that is optimized to discern cerebral cortex from surrounding structures and as such, they can not typically

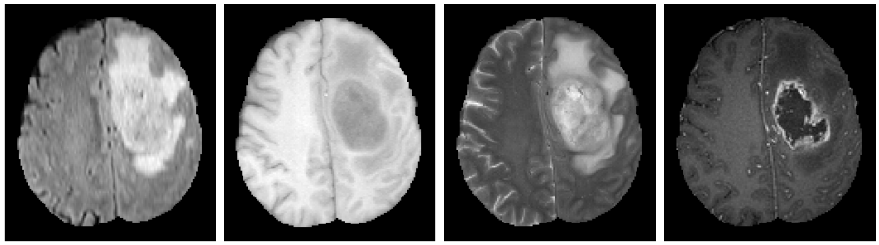


**Figure 1.1:** A scan of a tumor patient using four different sequences. From left to right: a FLuid-Attenuated Inversion Recovery (FLAIR) sequence which shows the full extent of the tumor (core and edema), a T1-weighted scan which shows the tumor core darker compared to edema, a T2-weighted scan which shows the texture of the core and a Gadolinium-enhanced T1-weighted scan which shows the necrotic (or fluid-filled) part of the core as dark and the part of the core where the blood-brain barrier has been broken as bright.

handle the large variability in brain MR imaging data as acquired in the clinical setting. Thus clinicians are forced to visually inspect the two-dimensional slices of the 3D volume, or manually delineate structures of interest from the scans. This introduces rater-dependent variability in the analysis which in turn reduces the accuracy and sensitivity of the results. In order to obtain reliable and repeatable segmentations of subcortical structures, essential for early diagnosis of many neurological and neuropsychiatric disorders, and assess if physical brain damage, e.g., tumors infarcts or lesions, might contribute to a patient's symptoms, there is an urgent need for computational methods that can readily handle the multi-contrast MR images that are acquired in routine clinical practice.

### 1.0.1 Goals of the project

In this PhD project the overall goal was to enlarge the scope of quantitative brain MRI analysis from mere scientific studies of the human brain into real-world clinical applications benefiting people suffering from devastating brain diseases. Given the extreme versatility of MRI and the lack of standard acquisition protocols for imaging the brain in clinical settings, we attempted to develop tools that can robustly analyze scans with various number of contrasts, as well as scans of patients with brain pathologies. To achieve this goal, the project was divided into two separate steps.

**Step 1** The first part of the project was dedicated to developing and validating an adaptive tool for automated brain parcellation of MRI data of healthy subjects into 39 different cortical and sub-cortical structures. We first developed a segmentation framework for single-contrast MR data, and then extended this framework to multi-contrast scans. Validation was done on multiple data sets in order to show that the proposed tool can readily handle all kinds of different data that might be encountered in clinical practice.

**Step 2** In the second part of the project we integrated models of pathology into the healthy brain parcellation framework resulting in a tool that simultaneously detects white matter lesions related to multiple sclerosis and segments the surrounding neuroanatomy. Performance of the method was tested on a benchmark data set and compared to state-of–the-art lesion detection methods.

### 1.0.2 Overview of the thesis

The rest of the thesis is divided as follows:

- Chapter 2 offers a brief overview of how the field of automated MR brain segmentation has developed and what type of methods are currently being used. We further discuss the main limitations of these methods, and relate the modeling approach taken in this project to previous work and to the still existing problems in whole-brain segmentation.
- Chapter 3 introduces the whole-brain segmentation framework which has been developed in this thesis. We build upon the work presented in [VL09], which we have extended and brought into practical applications. Once the modeling framework has been laid down, we overview the results from papers A and B and discuss the findings.
- Chapter 4 extends the whole-brain segmentation framework by introducing models for white matter lesions. The chapter starts with a brief overview into different methods that have been previously suggested for lesion segmentation, followed by a discussion of some of the shortcomings of these approaches. Next, we describe how the framework presented in chapter 3 is extended to joint whole-brain and lesion segmentation, and finally we overview and discuss the main findings from paper C.
- Chapter 5 provides a conclusion by summarizing the main contributions of the work that has been conducted during this project. This chapter also provides a brief overview of the other projects that I have been involved in, but which were not the main focus of the PhD project.

- Finally chapter 6 discusses some potential avenues for future work based on the contributions presented in this thesis.

The general aim of this thesis is to give the reader further insight into the developed models, and to discuss the main up- and downsides of the proposed approaches.

Chapter 2

# Overview of current approaches to whole-brain segmentation

This chapter is meant as a short overview of some of the most popular methods used in whole-brain segmentation. The methods listed are a small subset of those available. The idea is to give the reader an understanding of how the field has developed over the years, and where there is still room for improvement. The chapter is constructed as follows:

- First, we introduce the segmentation problem, and motivate why automated methods are needed.
- Next, we overview two different modeling approaches which most segmentation algorithms build upon, and provide concrete examples of both approaches in the form of segmentation tools that have been used as benchmark methods during the course of this project.
- Then, we point to some problems in the current approaches, and some solutions that have been suggested to alleviate these problems.
- Finally, we relate our segmentation method, which is presented in chapter

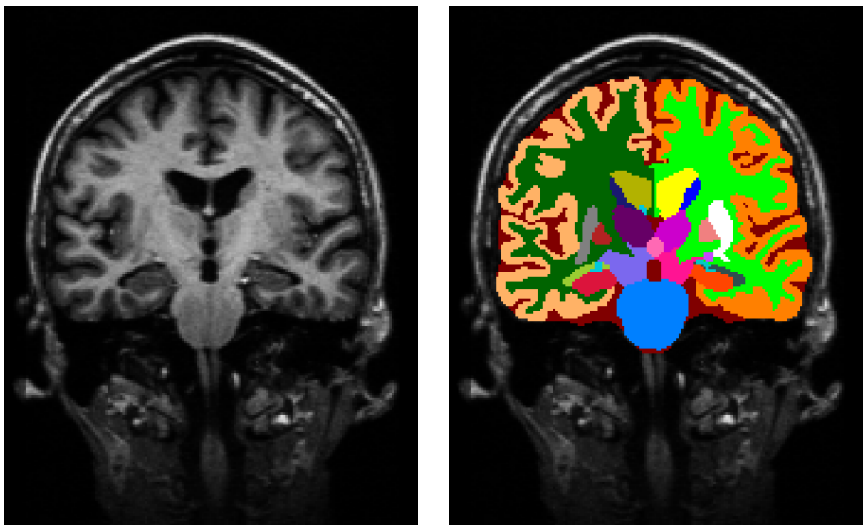


**Figure 2.1:** Left: a T1-weighted MR scan. Right: a corresponding segmentation into 39 different neuroanatomical structures done manually by an expert radiologist.

3, to the state-of-the-art, and discuss why our approach might be beneficial.

## 2.1 Introduction

So-called *whole-brain segmentation* is the task of assigning a neuroanatomical label to each voxel in an MR image. Typically this task is done by a trained radiologist who manually assigns each voxel in a 3D MR scan to one of (possibly) many neuroanatomical labels. Figure 2.1 shows an example slice from an MR scan and the corresponding manual labeling done by an expert. In this particular case the manual labeling protocol included 39 different neuroanatomical structures a single voxel can be assigned to.

These segmentations of different brain structures are useful, and sometimes necessary, for a multitude of clinical and research applications. An example situation where accurate segmentations are required is radiation therapy planning, where the locations of the radiated area, typically a tumor, and the surrounding structures need to be known precisely in order to target the radiation dose so that healthy brain structures are not affected. Another example is finding

suitable biomarkers for predicting and tracking the development of different central nervous system disorders. This has been successfully done in the case of Alzheimer's disease (AD), where biomarkers based on the volumes of different brain regions have been shown to be sensitive in discriminating between healthy and AD subjects, and also in predicting if a subject with questionable AD will later on progress to full onset of AD [FSB+02]. However, acquiring these segmentations manually is a painstaking task, which can take up to a week for a single 3D MR scan [FSB+02]. The amount of MR scans produced in the clinics and neuroscientific studies[1] nowadays, has made development of automated tools for brain segmentation a necessity.

Due to the complex anatomy of the human brain and the versatility of the MR imaging modality, devising a general all-purpose segmentation tool suitable for all kinds of scans is far from trivial. The first problem we are faced with is that many of the different neuroanatomical structures have very similar intensity properties [FSB+02], and thus can not be segmented based solely on their appearance in MR scans. The radiologists who manually segment these images, often have years of experience and detailed knowledge of the human neuroanatomy, which allows them to accurately delineate the structures. The second problem is that, as noted in the motivation chapter, in a typical clinical MR scan session multiple images with different contrast-properties are acquired which all highlight different properties of the brain. Ideally, the automated tool should be readily able to analyze all these different scans, implying that such a tool should be robust to changes in the number and contrast of the input MR scans. Next we will give a brief overview of some of the current, and previous, approaches that have been suggested for automating the task of whole-brain segmentation, and discuss why these might still not be considered a final solution.

## 2.2 Whole-brain segmentation: from tissue classification to multi-atlas labeling

We will divide the segmentation models into two different main categories depending on whether the underlying model is *parametric* or *non-parametric*. Although other categorizations are also possible – one could at least think of dividing the methods to supervised and unsupervised, or generative and discriminative approaches – we feel that the chosen categorization is most suitable within this particular project. However, common to all approaches is that they

[1] For example in relation to the Alzheimer's Disease Neuroimaging Initiative (ADNI) study, thousands of patients have been scanned and that is only one study!

rely on a training set consisting of MR scans and their manually obtained segmentations (such as is shown in figure 2.1) which are used for learning the models.

### 2.2.1 Parametric models

The earliest automated segmentation methods aimed at labeling the three main tissue classes (white matter, gray matter and cerebro-spinal fluid) in the brain. These *tissue segmentation* methods often relied on parametric *generative* models, where the available training set is summarized in relevant statistics, which are then used to inform the segmentation of new target MR scans. These models were defined to be generative as they allowed for generating synthetic MR data by sampling from the model. The first approaches were quite simple, typically assigning each voxel to each tissue class based solely on intensity information. One simple example of such a method [BHC93] is a Bayesian classifier based on a Gaussian mixture model (GMM), where the mixture parameters summarize the relevant information from the training set. Each Gaussian in the mixture was associated with one of the tissue classes, and the parameters were learned as the maximum likelihood (ML) estimates from the training data. Once the classifier was trained, each voxel in a target scan was simply labeled by computing the probabilities of belonging to each of the Gaussians and assigning the label with the highest probability to the voxel. This approach represents a *supervised* model as the relation between the tissue labels and intensities is learned from the training data set.

The next key development in tissue segmentation was to use the Bayesian modeling framework to include information about the spatial layout of tissue classes in the form of a prior distribution, typically called a *probabilistic atlas* [VMVS99b, AF97]. Such an atlas was constructed by computing the frequency with which a tissue class was present in each voxel across the training labelings. The probabilistic atlas represented the prior knowledge of seeing a specific tissue class in a given voxel before any data is observed. This prior information was then combined with an unsupervised intensity model, where the intensity of each tissue class was modeled with a single Gaussian distribution [VMVS99b, AF97]. A target MR scan was then labeled as before, but using the tissue probability maps to weigh the Gaussian intensity model. An important point to make here, is that the incorporation of the probabilistic atlas allowed for learning the Gaussian parameters in a clustering manner *from* the target scan. Thus the intensity model is defined to be *unsupervised* as a specific relation to the training data intensities is not established. Considering a clinical application this is a very desirable property as it makes the method contrast-adaptive [VMVS99b, AF97]. Furthermore, including multi-contrast

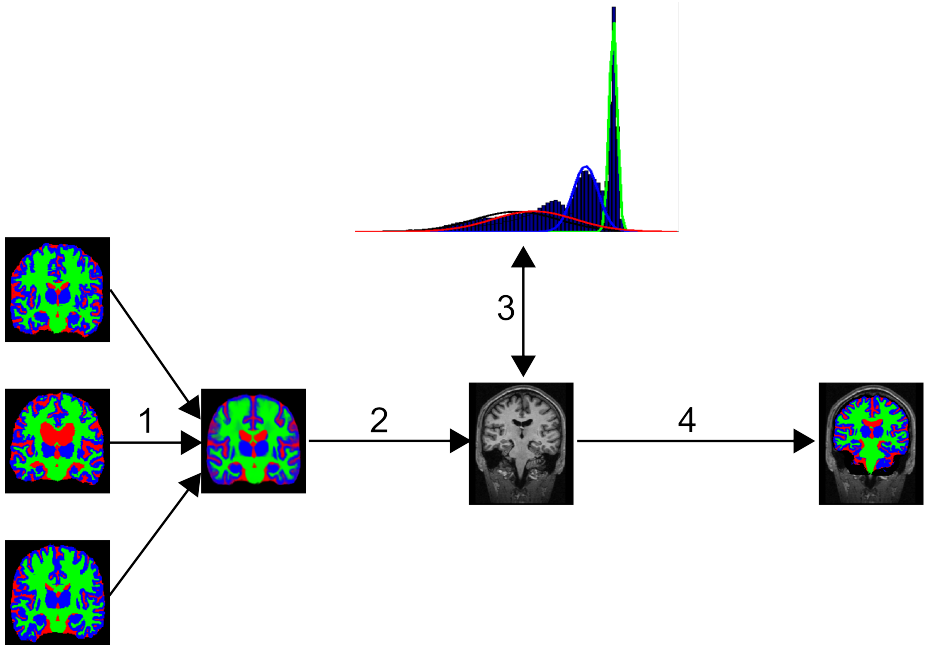



**Figure 2.2:** A schematic illustration of the segmentation pipeline of the tissue classification approaches. 1. A probabilistic tissue atlas is constructed from training scans. 2. The atlas is aligned with the target scan. 3. A GMM is fitted to the target scan voxel intensities based on the spatially varying prior tissue weights. 4. Final labeling is produced given the prior and GMM distributions.

scans is easily achieved by using multivariate Gaussian distributions. With this framework, a target image was processed in three steps: first the probabilistic atlas was co-registered to the target image, next the Gaussian parameters were learned, and finally a labeling was produced. This process is visualized in figure 2.2.

Most of the parametric models currently used for *whole-brain* segmentation still build upon the framework used in tissue segmentation. However, as the distinct neuroanatomical structures have very similar intensity characteristics [FSB$^+$02], these methods typically use very detailed prior information of the expected shape and relative positioning of different brain regions, using surface-based [KSG98, PFS$^+$03, PSKJ11, CET98] or volumetric [FSB$^+$02, PFG$^+$06] models. The prior models of anatomy are then combined with supervised intensity models, very similar to the simple Bayesian classifier described above, which encode the typical intensity characteristics of the relevant neuroanatomical structures. The

intensity characteristics of the different structures can be learned from the training data either for each individual voxel [FSB+02, PFG+06] or for the entire structure class [KSG98, PFS+03, PSKJ11, CET98].

To give an example of a parametric whole-brain segmentation approach, next we will briefly describe the segmentation model behind a very popular segmentation tool called **FreeSurfer** [FSB+02], which we have used as one of the benchmark methods during this PhD project.

#### 2.2.1.1 FreeSurfer

FreeSurfer builds on a generative segmentation approach, which uses a probabilistic atlas of the neuroanatomy in combination with a supervised intensity model based on Gaussian distributions. The probabilistic atlas is learned from expert labelings in the same manner as the tissue probability maps, only now instead of three tissue classes the number of possible structure labels is much higher. To model the structure intensities, each voxel and structure label is associated with a Gaussian distribution where the mean and variance are learned from the training data. In particular, at a given voxel the mean for a structure class is calculated as the mean intensity over the training scans where the class was present in that voxel. Given the mean, the variance is computed in a similar manner. To ensure spatial smoothness of the segmentations, FreeSurfer also uses a Markov random field (MRF) prior which encourages neighboring voxels to have the same label. Segmentation of a target scan is then obtained by first aligning the probabilistic atlas to the target scan using a non-linear registration approach, and then searching for a labeling which best satisfies both the intensity and the anatomical constraints.

### 2.2.2 Non-parametric models

In contrast to the parametric segmentation models, the *non-parametric* methods do not summarize the training data into a set of parameters but instead use each of the training data points for segmenting the target image. These methods have recently become arguably the most popular segmentation paradigm, and are typically implemented in the form of multi-atlas label fusion[2] [RBMMJ04, HHA+06, ISR+09, AMoBOdS09, SYVL+10, RRMJ04, WSD+13, CMF+11, RHS11, TWC+13, WWZ+13, AL13, ZGC14]. The main idea is that, given a training set of $M$ MR scans and expert labelings, each of the training MR scans is first registered to

[2]Note, however, that early implementations used *a single-atlas* approach with only a handful of manually delineated structures [DHT+99]

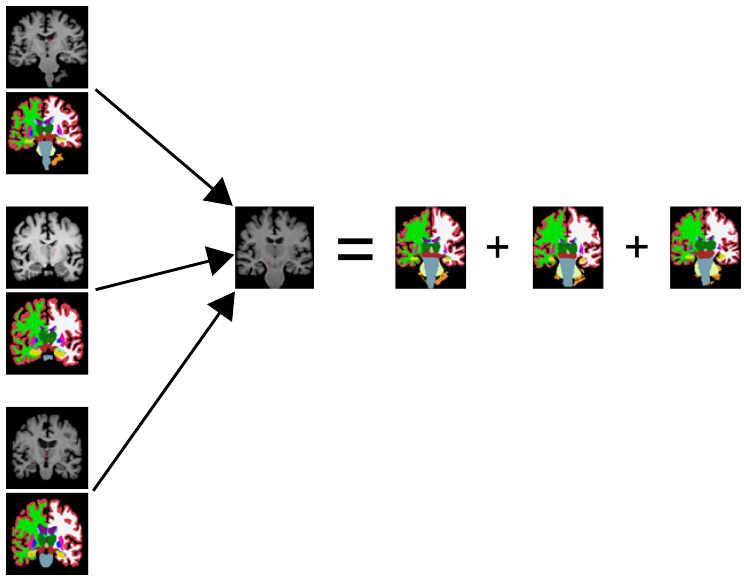


**Figure 2.3:** A schematic illustration of the multi-atlas label fusion approaches. Each of the three training scans is registered to the target scan resulting in three candidate segmentations.

the target scan, and then the training labelings are warped into the target scan space using the learned transformations. This procedure will result in $M$ possible labelings of the target MR scan, see figure 2.3 for an illustration. Finally the candidate labelings are fused to obtain the final segmentation. In summary, the segmentation of a new image consists of two steps: **pair-wise registration** and **label fusion**.

The multi-atlas methods have become very popular for three main reasons: first, they have been shown to give high segmentation accuracies [LW12], second, there is a large selection of freely available registration algorithms (see [KAA+09] for an overview and evaluation of 14 non-linear registration algorithms), and finally the approach, apart from registration, can be quite easily implemented. Given that the choice of registration method is somewhat a matter of preference[3], the main difference between the approaches lies in the label fusion step.

The first, and arguably the most simple, approach to fuse the labels into a single segmentation was majority voting [RBMMJ04, HHA+06], where each voxel is assigned the most frequent neuroanatomical label across the candidate

[3] Although many multi-atlas methods use the registration approach described in [AEGG08], which was one of the best performing methods in the evaluation performed in [KAA+09]

segmentations. Even though majority voting relies on a very simple fusion strategy, it has been shown to yield good results [RBMMJ04, HHA+06] and is still often used. The downside of majority voting is that, if many of the training subjects are poorly registered to the target scan they still "have a vote" in the final segmentation. This naturally can have a negative impact on the resulting final segmentation. Recent developments have been aimed at alleviating this problem by weighing the candidate labelings based on intensity information between the training and target MR scans after the pair-wise registrations. The training scans that closely resemble the target scan are assigned a higher weight in the voting, thus reducing the effect of poorly registered training subjects. The weighting can be done globally [AMoBOdS09], where all voxels in a candidate labeling are assigned the same global weight, or locally [AMoBOdS09, SYVL+10, ISR+09], where each voxel in a training labeling gets a different weight. Other strategies, apart from the voxel-wise and global voting, for fusing the labels have been suggested as well (see for example [RRMJ04, CMF+11, RHS11, TWC+13, WWZ+13, AL13, ZGC13]), but most of them still rely on weighting based on the intensity similarities.

Again to give a concrete example, we briefly describe the theory behind two very successful multi-atlas approaches, which have been shown to yield very accurate results and have been used as benchmark methods during this PhD project.

#### 2.2.2.1 BrainFuse

BrainFuse [SYVL+10] uses intensity-weighted label fusion, but formulates the problem in a generative probabilistic framework. It assumes that each voxel in the target scan is generated from one of the candidate segmentations. This results in a membership field over the target image, which indicates which candidate labeling generated each voxel in the target scan. Smoothness of this membership field is enforced with an MRF prior, which basically encourages that neighboring voxels are generated from the same candidate labeling. Intensity-based weighting between the target and training scan voxels is done using a Gaussian distribution, where a fixed variance is assumed for all voxels. Note, that this model is a generalization of some of the different approaches to label fusion listed above: if the MRF prior is not enforced and the variance of the Gaussian distribution is set to infinity, all the candidate labelings are assigned an equal weight, and the model reduces to majority voting. Similar to FreeSurfer, a segmentation is then obtained by looking for the labeling that best fulfils the model constraints given the target data.

#### 2.2.2.2 PICSL MALF

Most of the multi-atlas methods assign the voting weights to each candidate labeling independently. This relies on the assumption that the segmentation errors in the candidate labelings are different, and thus can be reduced by voting. However, if the segmentation errors are very similar the fused segmentation will still exhibit the same error. PICSL MALF [WSD+13] tries to account for these systematic errors by assigning the weights to the different candidate labelings jointly. The voting problem is formulated in terms of trying to minimize the total expected error between the unknown true labeling and the fused labeling in every voxel. To achieve this, the expected pairwise joint label differences between the training scans and the target scan are approximated using intensity similarity information. Once the weights have been computed, the target image is labeled by weighted voting. To further account for registration errors, PICSL MALF also performs a local search between each training scan and the target scan to refine the mapping between the images.

## 2.3 Main limitations in the current approaches

One of the main difficulties in applying the current whole-brain segmentation approaches, both parametric and non-parametric, in a general clinical setting, is that they are supervised, i.e., the target scans are assumed to have the same intensity properties as the training scans. This approach has two fundamental limitations: first, if the training and target scans come from different scanners or have been obtained with different scan-sequences, the segmentation performance often degrades due to the different contrast properties in the scans. In [HF07, RCP13] the authors show that this is true even when both the training and target data consist of T1w scans with similar contrast properties, but which have been acquired with different scanner platforms. Second, and more importantly, most of the research on whole-brain segmentation has been targeted for segmenting T1-weighted images, although, as mentioned before, in a clinical setting multiple images are typically acquired with different contrast, different resolution and providing complementary information. Concentrating solely on the T1w contrast hinders the translation of the promising research results into clinical use which should be considered the ultimate goal.

Some suggestions for extending the supervised segmentation approaches to work across different MR contrasts have been presented. The most straight-forward approach would be to expand the training data library to include all the possible scan-sequences and scanner types one might encounter in clinical applica-

tions. However, as pointed out in the introduction, this is likely not possible due to the versatility of MR and the laborious task of manually segmenting the images. Another approach, called histogram matching or intensity normalization [NUZ00, RCP13], is to "match" the intensity profile of the training data to that of the target data. This approach however only works if the training and target scans have been acquired with a similar sequence, i.e., both are T1-weighted scans acquired with different sequences or scanners. Recently, contrast synthesis [IKZ+13, RCP13] has been suggested as a solution. This approach goes around the problem by generating a new image from the target scan, where the intensity profile matches the training data and segments this "synthetic" image instead. However, this approach still requires a training set consisting of images scanned with both the target contrast and the contrast that we wish to synthesize.

One limitation related to the non-parametric approaches, is the high computational cost of performing the multiple pair-wise non-linear registrations between each of the training scans and the target scan. Note that the parametric methods only require a single registration which aligns the probabilistic atlas to the target scan, and thus are typically faster than the multi-atlas methods. Different solutions to alleviate the computational cost have been suggested. In [AHH+09], only the training subjects that are most similar to the target subject *before* registration are used. Alternatively this estimation can be done on-the-fly while computing the registrations [vRIA+10]. Recently in [CMF+11, TGCC14] a patch-based approach to label fusion was proposed. This approach relaxes the voxel-to-voxel correspondence criteria so that linear, as opposed to the costly non-linear, registrations can be used. A slightly different approach to multi-atlas segmentation is taken in [ZGC14], where the authors train a separate random atlas forest from each MR scan and labeling in the training data. The candidate labelings from each random forest are then fused using majority voting to produce a final segmentation. This approach yields very fast segmentations, because it relies on patch-based classification. However, typically the best segmentation results are obtained using non-linear registration tools [LW12].

A more subtle difficulty in applying the non-parametric methods to clinical practice is how to deal with pathologies. Often in research setups, and especially when validating the segmentation tools, only healthy subjects are used, although in the clinics the opposite is true. In the parametric approaches, as we will see in chapter 4, models of pathologies can be easily included into the framework, whereas for the non-parametric approaches it is not quite clear how this should be done. The main problem is the registration step, which becomes very difficult due to the random location and appearance of pathologies such as lesions and tumors.

**Extending the scope of tissue classification** In the next chapter we present a segmentation framework, which tries to address the problems described above. In contrast to the aforementioned supervised approaches to whole-brain segmentation, we build upon the unsupervised approaches which are typically used in tissue segmentation [VMVS99b, AF97]. As such, this approach readily handles contrast changes and multi-contrast scans, and is also computationally less demanding compared to the non-parametric approaches.

The method is closely related to the works presented in [AF05] and [BP08]. However, in [AF05] only tissue classification was attempted, whereas in [BP08] the authors rely on a supervised intensity initialization approach, and segment only a handful of different structures.

Chapter 3

# Whole-brain segmentation using a generative modeling framework

This chapter focuses on the first part of the PhD project, which deals with segmenting the brain into a multitude of cortical and sub-cortical structures. The chapter is divided as follows:

- The first section introduces the generative parametric segmentation model developed in this thesis. We describe in detail the main two components of the model: the probabilistic prior of human neuroanatomy and the model of MR intensities.
- The second section describes how to do inference, i.e., how a target MR scan is labeled, using the proposed framework.
- The third section briefly overviews the main experiments and results from papers A and B.
- The fourth section concludes the chapter with a discussion of the results.

## 3.1 Modeling framework

As stated in the previous chapter our segmentation task consists of finding a labeling $\mathbf{l} = \{l_1, \ldots, l_I\}$ given a (possibly) multi-contrast target MR scan $\mathbf{D} = \{\mathbf{d}_1, \ldots, \mathbf{d}_I\}$ with $I$ voxels. Here the label in each voxel can take on one of $K$ possible classes, i.e., $l_i \in \{1, \ldots, K\}$, and the vector $\mathbf{d}_i = (d_i^1, \ldots, d_i^N)^T$ contains the intensities of each of the available $N$ contrasts in voxel $i$.

Similar to all the parametric whole-brain segmentation models, our segmentation model consists of two parts: a **segmentation prior**, $p(\mathbf{l})$ which is a probability distribution over the possible labelings and encodes the spatial location and shape of different neuroanatomical structures. The second part of the model, is a **likelihood function**, $p(\mathbf{D}|\mathbf{l})$, which translates the different structure labels into intensities of the target scan. This type of model is defined to be generative, as it allows us to generate new synthetic MR scans by first sampling a labeling $\mathbf{l} \sim p(\mathbf{l})$ from the prior distribution and then the data from the likelihood function, which is conditioned on the labeling, $\mathbf{D} \sim p(\mathbf{D}|\mathbf{l})$. In order to estimate the labeling $\mathbf{l}$ given target data $\mathbf{D}$, we "invert" the model, i.e., we try to infer the most probable segmentation given the input data. The graphical representation of the full generative model is shown in figure 3.1.

Within the defined generative framework, "inverting" the model is achieved by writing out the posterior probability of a labeling given the target scan using Bayes' rule:

$$p(\mathbf{l}|\mathbf{D}) = \frac{p(\mathbf{D}|\mathbf{l})p(\mathbf{l})}{p(\mathbf{D})}. \tag{3.1}$$

Once we can estimate the posterior probability for all possible labelings, we are finished with the segmentation task and the sought after labeling is obtained as the maximum-a-posteriori (MAP) estimate from eq. 3.1[1]. However, in order to write out the segmentation posterior, we need to detail the actual parametric form of our prior and the likelihood distributions.

### 3.1.1 Segmentation prior

As the segmentation prior $p(\mathbf{l})$ we use a generalization of the probabilistic brain atlases, which have been typically used in MR brain segmentation [AF97, VMVS99b, VMVS99a, VMVS01, ZFE02, FSB+02, AF05, PGLG05, PFG+06, DMVS06, ATWF06, PBN+07]. Instead of computing the prior probabilities as the frequency with which each structure is observed in each voxel in the training

[1] Although in general finding the MAP estimate might be a difficult task in itself.

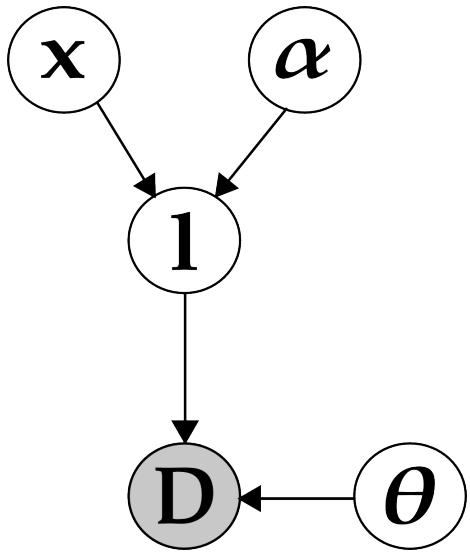



**Figure 3.1:** Graphical model of the generative segmentation framework. The parameters $\boldsymbol{\alpha}$ and $\mathbf{x}$ relate to the segmentation prior, whereas $\boldsymbol{\theta}$ collects the likelihood function parameters. The target data $\mathbf{D}$ is observed which is denoted by the shading.

labelings, we parametrize our prior distribution as a mesh with $V$ vertices. Each vertex in the mesh has an associated probability vector of length $K$, where each element specifies the probability for each of the $K$ labels of occurring around the vertex. The resolution of the mesh, i.e., the amount of mesh vertices, is locally adaptive, being sparse in large uniform regions and dense around the structure borders allowing for a compact encoding of the human neuroanatomy. The model was first introduced in [VL09], and in this project we have further developed it and applied it into practice.

In the following, we will briefly describe the generative model underlying the prior, and then detail how the parameters of the model are learned given a set of manual labelings. These derivations follow the ones presented in [VL09, IAN+15].

#### 3.1.1.1 The segmentation prior – a generative model of label images

In essence, given a set of $M$ manual labelings, we want to find the probability of seeing a label image $\mathbf{l}$ given the manually annotated examples: $p(\mathbf{l}|\{\mathbf{l}^m\})$. To

do this, we first set up a generative model of how label images are formed. For $M$ label images, the generation process proceeds in three steps:

1. A tetrahedral mesh over the image domain is defined, with node positions $\mathbf{x}^r$, referred to as the reference position, and connectivity $\kappa$, which details the topology of the mesh. Each mesh node $v$ is assigned a probability vector $\boldsymbol{\alpha}_v = \{\alpha_v^1, \ldots, \alpha_v^K\}$ which satisfy $\alpha_v^k \geq 0$ and $\sum_{k=1}^K \alpha_v^k = 1$. As we have no preference on the reference position, connectivity or structure probabilities in the mesh nodes, we assign uniform priors on these parameters, i.e., $p(\mathbf{x}^r) \propto 1$, $p(\kappa) \propto 1$ and $p(\boldsymbol{\alpha}) \propto 1$.

2. Given the reference position and the connectivity, $M$ deformed meshes are sampled from a deformation prior defined as:

$$p(\mathbf{x}^m|\mathbf{x}^r, \kappa, \beta) \propto \exp\left(-\beta \sum_{t=1}^{T} U_t^\kappa(\mathbf{x}^m, \mathbf{x}^r)\right), \tag{3.2}$$

   where $\mathbf{x}^m$ denotes the node positions of the $m$th deformed mesh, $T$ is the number of tetrahedra in the mesh, $U_t^\kappa(\cdot)$ is a penalty for deforming tetrahedron $t$ from its reference position to its actual position, and $\beta$ is a scalar that controls the stiffness of the mesh, i.e., how large deformations are allowed. As a deformation penalty we use the one proposed in [AAF00], which goes to infinity when the Jacobian determinant of the deformation approaches zero. This choice prevents the mesh from tearing or folding onto itself. In this work we have assumed a fixed value for beta, which is set to 0.1.

3. Given a deformed mesh with node positions $\mathbf{x}^m$, the probability for observing a label $k$ at voxel $i$ is given by:

$$p_i(k|\mathbf{x}^m, \boldsymbol{\alpha}, \kappa).$$

   Because the node positions do not necessarily coincide with voxel locations the label probabilities at the voxels need to be interpolated. For voxel $i$, this is achieved by first identifying the tetrahedron that contains the voxel, and then using barycentric interpolation of the label probabilities at the vertices of the tetrahedron (for details see [VL09]). Finally, assuming conditional independence of the labels in the different voxels given the node positions, probability vectors and mesh connectivity, we have:

$$p(\mathbf{l}^1, \ldots, \mathbf{l}^M|\mathbf{x}^1, \ldots, \mathbf{x}^M, \boldsymbol{\alpha}, \kappa) = \prod_{m=1}^{M} p(\mathbf{l}^m|\mathbf{x}^m, \boldsymbol{\alpha}, \kappa), \quad \text{where} \tag{3.3}$$

$$p(\mathbf{l}^m|\mathbf{x}^m, \boldsymbol{\alpha}, \kappa) = \prod_{i=1}^{I} p_i(l_i^m|\mathbf{x}^m, \boldsymbol{\alpha}, \kappa) \tag{3.4}$$

The entire generative process is illustrated in fig. 3.2.

Having defined the model of label images, we can re-write the probability of a labeling given the training examples as:

$$p(\mathbf{l}|\{\mathbf{l}^m\}) = \int_{\boldsymbol{\alpha},\kappa,\mathbf{x}^r} p(\mathbf{l}|\boldsymbol{\alpha},\kappa,\mathbf{x}^r)p(\boldsymbol{\alpha},\kappa,\mathbf{x}^r|\{\mathbf{l}^m\})\mathrm{d}\boldsymbol{\alpha}\mathrm{d}\kappa\mathrm{d}\mathbf{x}^r. \tag{3.5}$$

Here we can further write the first term in the integral as:

$$p(\mathbf{l}|\boldsymbol{\alpha},\kappa,\mathbf{x}^r) = \int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x},\boldsymbol{\alpha},\kappa)p(\mathbf{x}|\mathbf{x}^r,\kappa,\beta)\mathrm{d}\mathbf{x}. \tag{3.6}$$

Computing the integrals in eq. 3.5 is not possible in practice. However, we can approximate the integrations using the *empirical Bayes approximation*, where we assume that the posterior distribution of the parameters, i.e., the reference position, connectivity and probability vectors, given the training data is highly peaked around its mode:

$$p(\boldsymbol{\alpha},\kappa,\mathbf{x}^r|\{\mathbf{l}^m\}) \approx \delta(\boldsymbol{\alpha}-\hat{\boldsymbol{\alpha}},\kappa-\hat{\kappa},\mathbf{x}^r-\hat{\mathbf{x}}^r,),$$

where $\delta(\cdot)$ denotes Dirac's delta function and the optimal parameter values are obtained by maximizing $p(\boldsymbol{\alpha},\kappa,\mathbf{x}^r|\{\mathbf{l}^m\})$. This reduces the expression in eq. 3.5 to:

$$p(\mathbf{l}|\{\mathbf{l}^m\}) \approx p(\mathbf{l}|\hat{\boldsymbol{\alpha}},\hat{\kappa},\hat{\mathbf{x}}^r),$$

which then allows us to apply eq. 3.6 in practice. Note that the learned optimal model parameters now summarize the relevant features of the training data, which is equivalent to the approach all the segmentation approaches based on parametric models take.

#### 3.1.1.2 Bayesian inference - estimating the reference position, connectivity and probability vectors

**Estimating $\boldsymbol{\alpha}$:** Assuming for now that the connectivity $\kappa$ and the reference position $\mathbf{x}^r$ are known, we learn the probability vectors $\boldsymbol{\alpha}$ along with each of the deformed node positions $\mathbf{x}^m$ as MAP estimates given the training labelings:

$$\begin{aligned}\{\hat{\boldsymbol{\alpha}},\{\hat{\mathbf{x}}^m\}\} &= \underset{\boldsymbol{\alpha},\{\mathbf{x}^m\}}{\operatorname{argmax}}\, p(\boldsymbol{\alpha},\{\mathbf{x}^m\}|\{\mathbf{l}^m\},\mathbf{x}^r,\kappa,\beta) \\ &\propto \underset{\boldsymbol{\alpha},\{\mathbf{x}^m\}}{\operatorname{argmax}} \prod_{m=1}^{M} \left[p(\mathbf{l}^m|\mathbf{x}^m,\boldsymbol{\alpha},\kappa)p(\mathbf{x}^m|\mathbf{x}^r,\kappa,\beta)\right] p(\boldsymbol{\alpha}),\end{aligned}$$

where Bayes' rule was used to obtain the second expression. The MAP estimates are found by optimizing the expression in an iterative fashion by first keeping

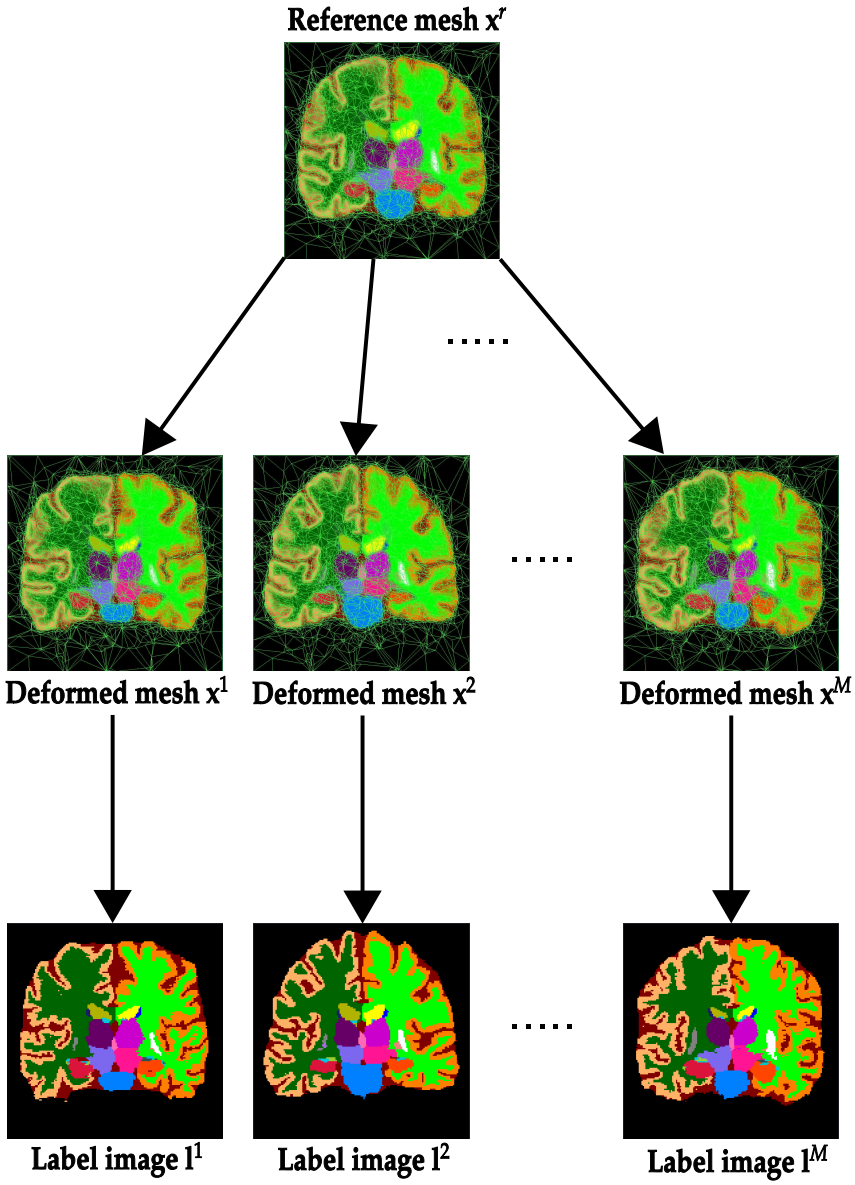



**Figure 3.2:** Illustration of the generative process underlying the prior model. First a mesh with reference position $\mathbf{x}^r$ and probability vectors $\boldsymbol{\alpha}$ is defined, next $M$ deformed meshes are sampled from the reference mesh, and finally label images are obtained by sampling from the interpolated voxel-wise probabilities.

the mesh node positions $\{\mathbf{x}^m\}$ fixed and updating $\boldsymbol{\alpha}$ using an Expectation-Maximization (EM) approach, and subsequently fixing $\boldsymbol{\alpha}$ and updating the mesh node positions using a conjugate-gradient optimizer. Updating the $\boldsymbol{\alpha}$ amounts to re-estimating the label probabilities at each location, whereas updating $\{\mathbf{x}^m\}$ performs a group-wise, non-rigid registration [IAN+15, VL09]. The exact form of the update equations can be found in [IAN+15, VL09].

**Learning the topology of the reference mesh:** As noted in [VL09], one potential problem with traditional voxel-wise probabilistic atlases, is that they are prone to over-fitting when a limited number of training data is available. In such cases, the atlas might assign a zero probability for a structure at a certain location only because the structure did not occur in that location in the training data. This problem is commonly handled by smoothing the atlas probabilities [Ash01] using, for example, a Gaussian smoothing kernel. However, in our mesh-based framework the over-fitting problem can be naturally handled by optimizing the mesh topology such that a proper amount of blurring is introduced [IAN+15, VL09]. Using a very sparse mesh, i.e., with a small amount of mesh vertices, leads to smoother probabilities as each vertex will model a larger spatial area. The optimal topology thus depends on the number of training labelings, such that smaller training sets yield sparser meshes.

Because we assumed uniform priors on the reference position $\mathbf{x}^r$ and the connectivity $\kappa$, we can estimate their optimal values by comparing different mesh topologies based on the so-called marginal likelihood, which is also known as evidence:

$$p(\{\mathbf{l}^m\}|\beta, \mathbf{x}^r, \kappa) = \int_{\boldsymbol{\alpha}} \left[ \prod_m \int_{\mathbf{x}^m} p(\mathbf{l}^m|\mathbf{x}^m, \boldsymbol{\alpha}, \kappa) p(\mathbf{x}^m|\mathbf{x}^r, \kappa, \beta) \mathrm{d}\mathbf{x}^m \right] p(\boldsymbol{\alpha}) \mathrm{d}\boldsymbol{\alpha}.$$

Again, the marginalizations over the parameters can not be performed in practice, but can be approximated in order to estimate the evidence of different reference mesh configurations (see [VL09] for details). In essence the evidence gives the probability for the labelings to have been generated from a reference mesh with vertex positions $\mathbf{x}^r$ and connectivity $\kappa$. This allows us to balance between model complexity, i.e., number of mesh vertices, and how well the training labelings are represented by the model.

Given these two optimization objectives, a full optimization of the prior model parameters then proceeds as follows: first, a reference mesh with high resolution and regular connectivity is placed over the image with the probability vectors in each node initialized to $1/K$. Next, the MAP estimates for the label probabilities alpha $\boldsymbol{\alpha}$ and the positions of each deformed mesh $\{\hat{\mathbf{x}}^m\}$ are computed, and finally the mesh topology is optimized. Optimization of the topology is performed

by randomly visiting each edge in the mesh and comparing the effect on the evidence of either keeping the edge while optimizing the reference position of the two nodes at its ends, or collapsing the edge into a single node and optimizing its reference position [IAN+15]. Further details on the optimization can be found in [VL09].

Figure 3.3 shows a fully trained mesh in its optimized reference position. Note the irregular size and number of the tetrahedra in different spatial locations of the image. The mesh is much finer along structure borders where more detail is needed, as compared to uniform areas which can be modeled with fewer mesh nodes, resulting in a compact representation of the human neuroanatomy.

#### 3.1.1.3 Summary of the segmentation prior

Once all the necessary parameters have been learned, we can write out the segmentation prior as:

$$p(\mathbf{l}|\hat{\boldsymbol{\alpha}}, \hat{\kappa}, \hat{\mathbf{x}}^r) = \int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x}, \hat{\boldsymbol{\alpha}}, \hat{\kappa}) p(\mathbf{x}|\hat{\mathbf{x}}^r, \hat{\kappa}, \beta) \mathrm{d}\mathbf{x},$$

where the hatted variables denote learned values. For the rest of the thesis, we drop the dependency on the learned variables and simply write:

$$p(\mathbf{l}) = \int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x}) p(\mathbf{x}) \mathrm{d}\mathbf{x}. \tag{3.7}$$

### 3.1.2 Likelihood

Next, we need to translate the label information encoded by the prior into intensities. This is achieved by the likelihood function, $p(\mathbf{D}|\mathbf{l})$ which gives us an explicit model relating each label to an intensity distribution. We parametrize the likelihood function as a mixture of Gaussian distributions, similar to [AF05]. The intensities of each structure, are thus modeled by multiple Gaussians[2], as opposed to the early tissue classification models where a single Gaussian was associated with each tissue class. The data in voxel $i$ for a structure class $k$ is generated from:

$$p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) = \sum_{g=1}^{G_k} w_{k,g} \mathcal{N}(\mathbf{d}_i|\boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}), \tag{3.8}$$

[2]Typically no more than three Gaussians are needed per structure

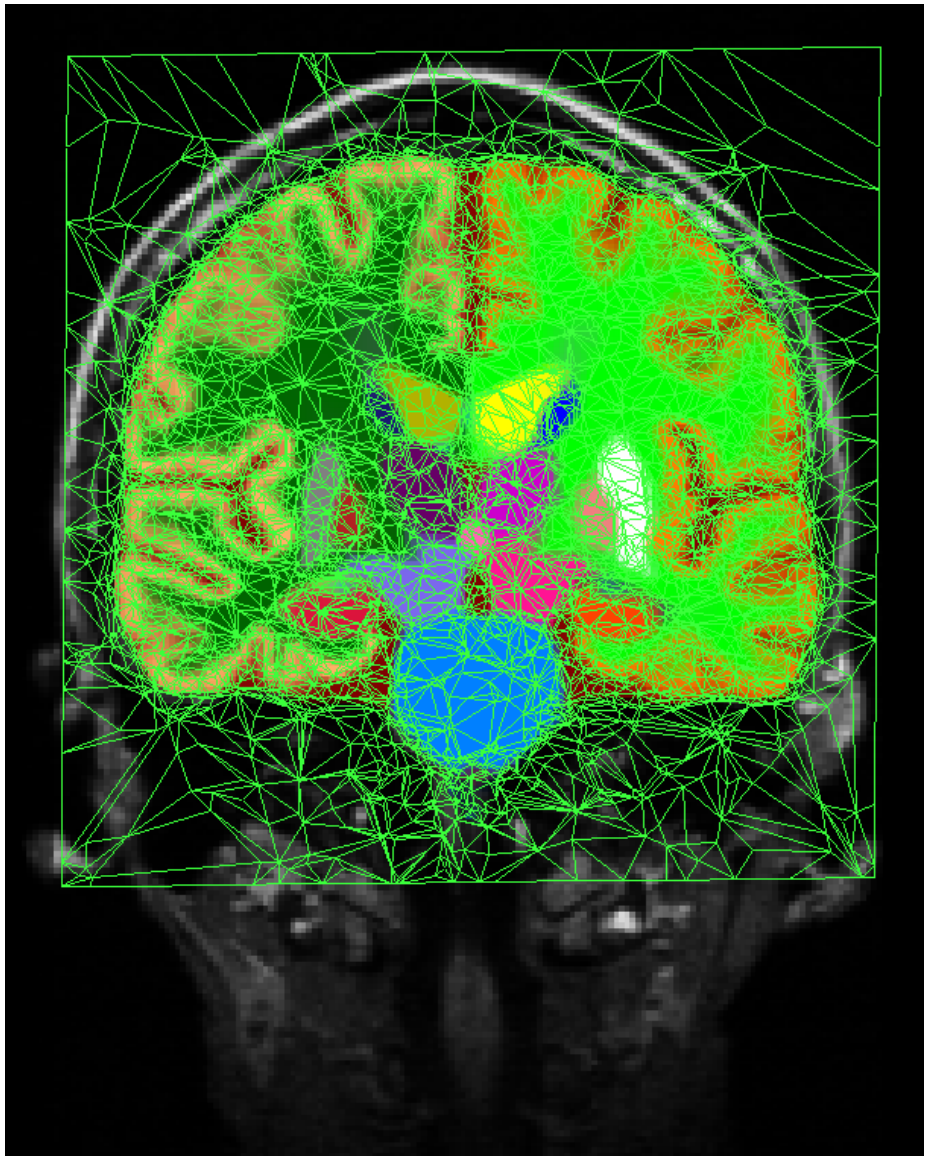


**Figure 3.3:** An optimal reference mesh learned from 20 training labelings.

where:

$$\mathcal{N}(\mathbf{d}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^N \det(\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}(\mathbf{d}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{d}-\boldsymbol{\mu})\right)$$

Here $G_k$ denotes the number of Gaussian distributions used for modeling the intensities of class $k$, $\boldsymbol{\theta} = \{w_{k,g}, \boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}\}$ are the weight, mean and covariance of mixture component $g$, and $\boldsymbol{\theta}$ denotes the set of all parameters. The weights are restricted to be larger than zero $w_{k,g} \geq 0$ and sum up to one $\sum_g^{G_k} w_{k,g} = 1$.

**Modeling intensity artifacts.** MR scans are corrupted by a smoothly varying intensity artifact typically called the *bias field*. This artifact is inherent to the imaging modality and appears as low-frequency multiplicative noise in the images [LIVL14]. Although the effect is present in all field strengths, it is more prominent in MR scans obtained using a high field strength scanner (e.g., 7 Tesla) [LIVL14]. Using a likelihood model that does not account for this effect, such as the GMM in eq. 3.8, results in degradation of segmentation results as shown in fig. 3.4.

In many segmentation approaches, for example in [FSB+02] and most of the non-parametric models covered in chapter 2, bias field correction is regarded as a pre-processing step. However, we will explicitly include it into the likelihood model similar to [WIG+96, VMVS99b, AF97]. The bias field is assumed to be a multiplicative and spatially smooth effect on the image intensities [WIG+96]. To facilitate computations, we use log-transformed image intensities and model the bias field as a linear combination of spatially smooth basis functions that are *added* to the local voxel intensities [VMVS99b]. The bias field term in voxel $i$ is expressed as:

$$\mathbf{b}_i = \mathbf{C}\boldsymbol{\phi}^i, \tag{3.9}$$

where

$$\mathbf{C} = \begin{pmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_N^T \end{pmatrix}, \quad \mathbf{c}_n = \begin{pmatrix} c_{n,1} \\ \vdots \\ c_{n,P} \end{pmatrix} \quad \text{and} \quad \boldsymbol{\phi}^i = \begin{pmatrix} \phi_1^i \\ \phi_2^i \\ \vdots \\ \phi_P^i \end{pmatrix}.$$

Here $P$ denotes the number of bias field basis functions, $\phi_p^i$ is the basis function $p$ evaluated at voxel $i$, and $\mathbf{c}_n$ holds the bias field coefficients for MR contrast $n$. In this work we model the bias field using a linear combination of cosine basis functions. Specifically, these are the P lowest frequency components of the Discrete Cosine Transform (DCT), where P is the number of basis functions per scan dimension, amounting to a total of $P^3$ basis functions over the full

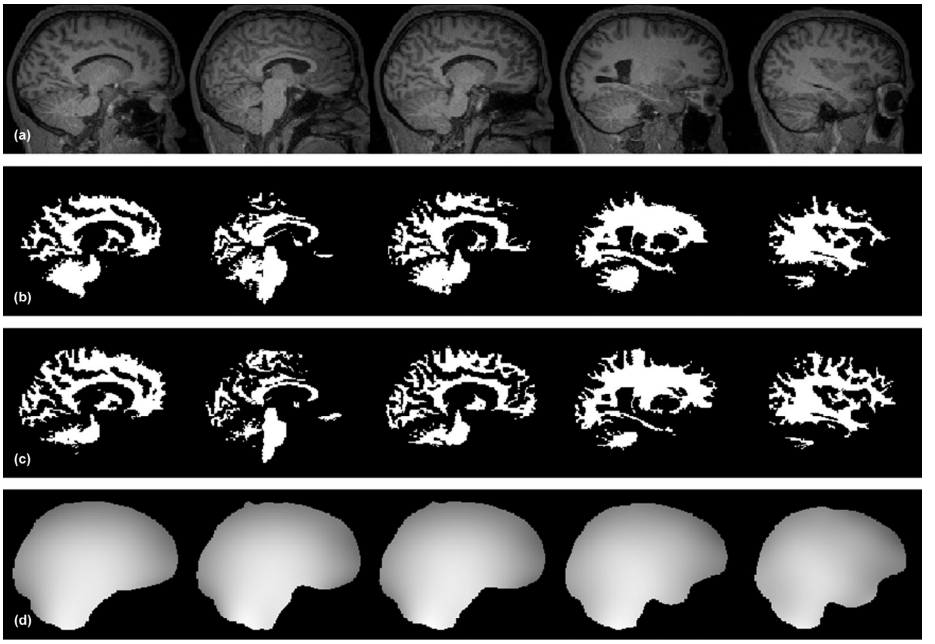



**Figure 3.4:** Illustration of the bias field effect: a) saggital slices from an MR scan of a single subject, b) white matter segmentation obtained without correcting for the bias effect, c) white matter segmentation when bias field correction is employed and d) the estimated bias field. Note the segmentation errors on the upper part of the brain when the bias effect is not removed. Figure from [VP15]

3D scan. Other parametrizations of the basis functions, such as B-splines or polynomials, are also possible [SZE98].

Including the bias field model into the GMM in eq. 3.8 results in:

$$p_i(\mathbf{d}_i|k,\boldsymbol{\theta}) = \sum_{g=1}^{G_k} w_{k,g}\mathcal{N}(\mathbf{d}_i - \mathbf{b}_i|\boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}), \tag{3.10}$$

where $\boldsymbol{\theta}$ now also collects the bias field coefficients $\mathbf{C}$. Given a labeling $\mathbf{l}$ and a set of parameters $\boldsymbol{\theta}$ the probability of a multi-contrast MR scan is given by:

$$p(\mathbf{D}|\mathbf{l},\boldsymbol{\theta}) = \prod_{i=1}^{I} p_i(\mathbf{d}_i|l_i,\boldsymbol{\theta}),$$

where we have assumed conditional independence between the voxels given the

labels. Finally the full likelihood function is written as:

$$p(\mathbf{D}|\mathbf{l}) = \int_{\boldsymbol{\theta}} p(\mathbf{D}|\mathbf{l},\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \tag{3.11}$$

In this work we have assumed a uniform prior on the likelihood function parameters, i.e., $p(\boldsymbol{\theta}) \propto 1$.

## 3.2 Inference

Having defined the prior and likelihood models, we can now write the segmentation posterior as:

$$p(\mathbf{l}|\mathbf{D}) \propto \left(\int_{\boldsymbol{\theta}} p(\mathbf{D}|\mathbf{l},\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}\right)\left(\int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{x}\right).$$

However, we are again faced with integrations that are not feasible to do in practice. To overcome this difficulty, we once more use the empirical Bayes approximation. Alternatively to the above expression, we can write out the segmentation posterior as:

$$p(\mathbf{l}|\mathbf{D}) = \int_{\mathbf{x}}\int_{\boldsymbol{\theta}} p(\mathbf{l}|\mathbf{D},\mathbf{x},\boldsymbol{\theta})p(\mathbf{x},\boldsymbol{\theta}|\mathbf{D})\mathrm{d}\mathbf{x}\mathrm{d}\boldsymbol{\theta}. \tag{3.12}$$

As in section 3.1.1.1, we assume that the posterior distribution of the parameters given the data is heavily peaked around its mode:

$$p(\mathbf{x},\boldsymbol{\theta}|\mathbf{D}) \approx \delta(\mathbf{x}-\hat{\mathbf{x}},\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}), \tag{3.13}$$

where the point estimates are given by:

$$\{\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}\} = \underset{\{\mathbf{x},\boldsymbol{\theta}\}}{\operatorname{argmax}}\, p(\mathbf{x},\boldsymbol{\theta}|\mathbf{D}). \tag{3.14}$$

Now we can approximate the segmentation posterior as:

$$\begin{aligned} p(\mathbf{l}|\mathbf{D}) &= \int_{\mathbf{x}}\int_{\boldsymbol{\theta}} p(\mathbf{l}|\mathbf{D},\mathbf{x},\boldsymbol{\theta})p(\mathbf{x},\boldsymbol{\theta}|\mathbf{D})\mathrm{d}\mathbf{x}\mathrm{d}\boldsymbol{\theta} \\ &\approx p(\mathbf{l}|\mathbf{D},\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}), \end{aligned} \tag{3.15}$$

which no longer involves intractable integrals. The resulting inference algorithm thus consists of two distinct steps: first, point estimates of the parameters are computed by maximizing eq. 3.14, and subsequently the segmentation is obtained by maximizing eq. 3.15.

**Computing the MAP parameter estimates.** Using Bayes' rule on the posterior distribution of the parameters given the data in eq. 3.14, we obtain:

$$\begin{aligned} p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{D}) &\propto p(\mathbf{D}|\mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}) p(\boldsymbol{\theta}) \\ &= \left( \sum_{\mathbf{l}} p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}) p(\mathbf{l}|\mathbf{x}) \right) p(\mathbf{x}) \\ &= \prod_{i=1}^{I} \left( \sum_{k=1}^{K} p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) p_i(k|\mathbf{x}) \right) p(\mathbf{x}), \end{aligned}$$

where the prior on the parameters disappears as it is assumed to be uniform. Taking the logarithm, we can rewrite the problem as the maximization of the following objective function:

$$\begin{aligned} \{\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}\} &= \underset{\{\mathbf{x}, \boldsymbol{\theta}\}}{\operatorname{argmax}} \, \mathcal{L}(\mathbf{x}, \boldsymbol{\theta}) \\ \mathcal{L}(\mathbf{x}, \boldsymbol{\theta}) &= \left[ \sum_{i=1}^{I} \log \left( \sum_{k=1}^{K} p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) p_i(k|\mathbf{x}) \right) + \log p(\mathbf{x}) \right]. \end{aligned} \tag{3.16}$$

We maximize the objective function using a coordinate-ascent approach, where the mesh vertex positions $\mathbf{x}$ and likelihood parameters $\boldsymbol{\theta}$ are iteratively updated, by alternately optimizing one while keeping the other fixed.

For optimizing the mesh vertex positions $\mathbf{x}$, we employ a standard conjugate-gradient (CG) optimizer [She94]. The gradient of the mesh node positions is given in analytical form:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{x}} = -\beta \sum_{t=1}^{T} \frac{\partial U_t^{\kappa}(\mathbf{x}, \mathbf{x}^r)}{\partial \mathbf{x}} + \sum_{i=1}^{I} \frac{\sum_k p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) \frac{\partial p_i(k|\mathbf{x})}{\partial \mathbf{x}}}{\sum_k p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) p_i(k|\mathbf{x})}. \tag{3.17}$$

Updating the vertex positions amounts to a registration process which deforms the probabilistic atlas to the target MR scan, based on the current segmentation estimate.

For optimizing the likelihood parameters $\boldsymbol{\theta}$ with fixed mesh node positions $\mathbf{x}$, we use a generalized expectation-maximization (GEM) algorithm [DLR77] similar to the one proposed in [VMVS99b]. GEM is well-suited for this problem, as we have a sum over the structure classes which can be considered "missing data" [Min98] (if the labeling was known there would be no segmentation problem). In the expectation step (E-step) the algorithm builds a lower bound, i.e., a local approximation to the objective function in 3.16, which touches the objective at the current parameter estimates, and in the maximization step (M-step) the lower bound is increased with respect to the parameters. Because the lower

bound touches the objective function, the algorithm is guaranteed to increase the objective at every iteration [DLR77, VMVS99b, Min98]. The E-step of the GEM algorithm involves computing the posterior probability for the GMM components of each structure class given the current parameter estimates and the data:

$$p_i(k^g|\mathbf{d}_i, \mathbf{x}, \boldsymbol{\theta}) = q_i^{k,g} = \frac{w_{k,g}\mathcal{N}\left(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i|\boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}\right) p_i(k|\mathbf{x})}{\sum_{k'=1}^{K} p_i(\mathbf{d}_i|k', \boldsymbol{\theta}) p_i(k'|\mathbf{x})}, \tag{3.18}$$

subsequently the parameters are updated given the current soft assignments as:

$$\boldsymbol{\mu}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i)}{\sum_{i=1}^{I} q_i^{k,g}}, \quad w_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}}{\sum_{i=1}^{I}\sum_{g'=1}^{G_k} q_i^{k,g'}},$$

$$\boldsymbol{\Sigma}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)^T}{\sum_{i=1}^{I} q_i^{k,g}},$$

$$\begin{pmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_N \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{A}^T\mathbf{S}_{1,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{1,N} \\ \vdots & \ddots & \vdots \\ \mathbf{A}^T\mathbf{S}_{N,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{N,N} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mathbf{A}^T\left(\mathbf{S}_{1,1}\mathbf{r}_{1,1} + \dots + \mathbf{S}_{1,N}\mathbf{r}_{1,N}\right) \\ \vdots \\ \mathbf{A}^T\left(\mathbf{S}_{N,1}\mathbf{r}_{N,1} + \dots + \mathbf{S}_{N,N}\mathbf{r}_{N,N}\right) \end{pmatrix}.$$

Here

$$\mathbf{A} = \begin{pmatrix} \phi_1^1 & \dots & \phi_P^1 \\ \vdots & \ddots & \vdots \\ \phi_1^I & \dots & \phi_P^I \end{pmatrix}$$

is a matrix collecting the $P$ basis functions evaluated at each voxel $i$. The term $\mathbf{S}_{m,n}$ is a diagonal matrix defined as:

$$s_{i,k,g}^{m,n} = q_i^{k,g}\left(\boldsymbol{\Sigma}_{k,g}^{-1}\right)_{m,n}, \quad s_i^{m,n} = \sum_{k=1}^{K}\sum_{g=1}^{G_k} s_{i,k,g}^{m,n}, \quad \mathbf{S}_{m,n} = \mathrm{diag}\left(s_i^{m,n}\right),$$

where each diagonal entry holds the sum over the precision matrix for each mixture component weighted by the soft assignments for each voxel. Finally $\mathbf{r}_{m,n} = (r_1^{m,n}, \dots, r_I^{m,n})^T$ is a vector denoting the residue image given by the difference between the original input data and the estimated bias corrected data as:

$$r_i^{m,n} = d_i^n - \frac{\sum_{k=1}^{K}\sum_{g=1}^{G_l} s_{i,k,g}^{m,n}\left(\boldsymbol{\mu}_{k,g}\right)_n}{\sum_{k=1}^{K}\sum_{g=1}^{G_k} s_{i,k,g}^{m,n}}.$$

**Computation of the final segmentation** Once the optimal point estimates for the parameters have been found, the approximate segmentation posterior can be written as:

$$p(\mathbf{l}|\mathbf{D}, \hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}) = \prod_{i=1}^{I} p_i(l_i|\mathbf{d}_i, \hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}).$$

Now the MAP segmentation can be easily obtained, by assigning the most probable label to every voxel independently as:

$$\hat{l}_i = \underset{k}{\operatorname{argmax}} \sum_{g=1}^{G_k} p_i(k^g|\mathbf{d}_i, \hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}).$$

### 3.2.1 Some notes on inference

The empirical Bayes approximation employed in order to estimate the integrations over the parameters naturally leads to a framework where the registration and segmentation are done jointly in an iterative manner. This is similar to the approach presented in [AF05] for tissue classification. As noted in [AF05], the joint framework, as opposed to sequential segmentation frameworks where registration and segmentation are considered two separate steps, is typically more involving to implement but can yield better results as the optimization task is actually completed. Furthermore, registration and segmentation can be thought of as two-sides of the same coin: having a perfect registration would solve, or at least simplify, the segmentation problem, whereas having a perfect segmentation would make registration much easier. Thus an iterative approach where one informs the other and vice versa, seems like a more fundamental and robust framework.

Technically we could use any optimization method in order to maximize the parameter posterior given the data. However, we have found in practice that the combination of CG and GEM works very well. For updating the positions of the mesh vertices, we also experimented with using a Levenberg-Marquardt optimizer but the CG optimizer gave better segmentation accuracies in our experiments. For optimizing the likelihood parameters, we have only used the GEM approach because it has two highly desirable properties: first, each iteration is guaranteed to increase the value of the objective function, although no guarantee is given that we will end up in the global maximum, and second, it does not require manual tuning of the size of the gradient step which is a typical problem in many other gradient-ascent optimizers. One thing to note is that in our case the update equations of the means, covariances and bias field coefficients are intertwined, i.e., the update step for the means depends on the current estimate for the bias field coefficients and vice versa. Thus

the lower-bound can not be maximized in one step as is the case for classical EM [DLR77, VMVS99b, Min98]. However, one could do multiple iterations between the update steps in order to exactly maximize the lower bound, but as the GEM-algorithm is guaranteed to increase the value of the objective function it is computationally more efficient to update the parameters only once and then recalculate the lower bound which is fast to compute in our case [VP15].

Finally, it is also instructive to note that the exact optimal values of the mesh vertex positions and the likelihood parameters are of no interest to us. Rather they are a nuisance that we can not get rid of because the marginalizations in eq. 3.12, are intractable to do. However, instead of the empirical Bayes approximation, we could also evaluate the segmentation posterior by sampling from the parameters. In this case the full segmentation posterior would be approximated by:

$$p(\mathbf{l}|\mathbf{D}) \approx \frac{1}{S}\sum_{s=1}^{S} p(\mathbf{l}|\mathbf{D}, \mathbf{x}(s), \boldsymbol{\theta}(s)),$$

where $\mathbf{x}(s)$ and $\boldsymbol{\theta}(s)$ are samples generated from the posterior distribution of the parameters given the data $p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{D})$. Having samples from the parameters would also allow us to put error bars on the segmentations, obtained by averaging over the samples of different structures. While the sampling approach was shown to give some improvement over the empirical Bayes approximation in [ISV13a], in this work we have used the analytical approximation which is computationally less demanding.

The nuisance parameters could also be treated in a variational inference framework as shown in [BCA15], where the intractable posterior distribution of the variables is approximated with a factorizable form. The resulting inference algorithm is a more general formulation of the EM algorithm which has a lower computational complexity than sampling approaches. The drawback is that the approximation to the true posterior will never be exact. Despite this, the authors show slightly increased segmentation performance compared to MAP segmentation. The improved results from both the sampling and the variational approaches indicate that a more Bayesian treatment of the segmentation problem seems to be beneficial.

## 3.3 Experiments and results

In this section we briefly present the results from papers A and B. We first describe the data sets used in the experiments, next we describe the experiments conducted to validate the proposed approach, and then provide a short overview

of the results from each experiment. The results are separately discussed in the final section of this chapter. The exact implementation details and cross-validation set-ups related to the experiments can be found in paper B and are not repeated here.

### 3.3.1 Data

We used five different MR data sets in the experiments: one data set was used *exclusively* for training the proposed and benchmark methods, whereas the four other data sets were used *only* for testing. This set-up ensures a fair comparison, and allows us to mimic a real clinical setting where the training and target data might come from different scanners, using different sequences and field strengths. In total the four test data sets consist of 203 MR scans, including 88 multi-contrast scans, which allows for rigorous testing of the proposed segmentation framework.

**Training data.** The training data set consists of 39 T1-weighted MR scans and corresponding manual segmentations, with 28 healthy subjects and 11 subjects suffering from questionable or probable Alzheimer's disease, and ages ranging from under 30 years old to over 60 years old [SYVL$^+$10]. The manual segmentations were performed by an expert radiologist using a protocol described in [CJFK89]. The scans were acquired on a 1.5T Siemens Vision scanner using an MPRAGE sequence with parameters: TR=9.7ms, TE=4ms, TI=20ms, flip angle = 10° and voxel size = $1.0 \times 1.0 \times 1.5\text{mm}^3$ (128 sagittal slices). The scan parameters were empirically optimized to maximize the gray-white matter contrast [BHP$^+$04]. An example scan and a corresponding manual segmentation from this data set are shown in the introduction of chapter 2 (see figure 2.1).

**Intra-scanner data.** The first test dataset consists of 13 T1-weighted scans acquired on a 1.5T Siemens Sonata scanner with the same sequence and parameters as the training data [HF07]. Given the similarity with the training data (vendor, field strength, pulse sequence), we refer to this dataset as the **"intra-scanner dataset"**. The manual segmentations were done by an expert radiologist using the same protocol as for the training data. An example scan and a corresponding manual segmentation from this data set are shown in figure 3.5.

**Cross-scanner data.** The second test dataset consists of 14 T1-weighted scans acquired on a 1.5T GE Signa Scanner using an SPGR sequence with parameters: TR = 35 ms, TE = 5 ms, flip angle = 45° and voxel size = $0.9375 \times 0.9375 \times 1.5\text{mm}^3$ (124 coronal slices) [HF07]. The manual segmentations were done by an expert radiologist using the same protocol as for the training data. We refer to this dataset as the **"cross-scanner dataset"**. An example scan and a corresponding manual segmentation from this data set are shown in figure 3.6.

**Multi-echo data.** The third test dataset consists of multi-echo FLASH scans from 8 healthy subjects acquired on a 1.5T Siemens Sonata scanner. The acquisition parameters were: TR = 20 ms, TE = 1 min, flip angle = 3°, 5°, 20° and 30°, and voxel size = $1.0\text{mm}^3$ isotropic [FSvdK$^+$04, ISV12]. The different flip angles correspond to different contrast properties, with the smallest angle having contrast similar to proton density (PD) weighting and the largest one having a contrast similar to T1-weighting. The manual segmentations were again obtained using the same protocol as for the training data. We refer to this dataset as the **"multi-echo dataset"**. A sample slice from this dataset, with flip angles 30° and 3°, is shown in figure 3.7.

**Test-retest data.** The fourth and final test dataset consists of 40 healthy subjects scanned at two different time points at different facilities, with scan intervals ranging from 2 days to six months, amounting to a total of 80 T1- and T2-weighted scans for the whole dataset [HLH$^+$12]. The scans were all acquired with 3T Siemens Tim Trio scanners using identical multi-echo MPRAGE sequences for the T1 and 3D T2-SPACE sequences for the T2, with voxel size = $1.2 \times 1.2 \times 1.2\text{mm}^3$. We refer to this dataset as the **"test-retest dataset"**. One of the scans had to be excluded because of motion artifacts. Moreover, some of the T2-weighted scans have minor artifacts not present in the T1-weighted scans. These scans were however included in the experiments. Manual segmentations were not available for this dataset; however, these scans are still useful in test-retest experiments quantifying the differences between the two time points. Ideally, as all the subjects are healthy, the biological variations should be small and the segmentations between the two time points should be identical. An example of the T1- and T2-weighted scans is shown in figure 3.8.

### 3.3.2 Experiments

We thoroughly tested our segmentation framework in five different experiments. First, we compared the segmentation accuracy and speed of the proposed ap-

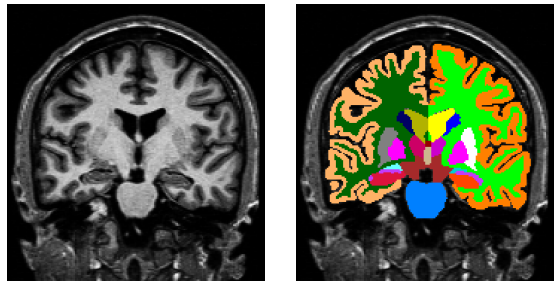


**Figure 3.5:** On the left an example slice from the intra-scanner dataset and on the right a corresponding manual segmentation.

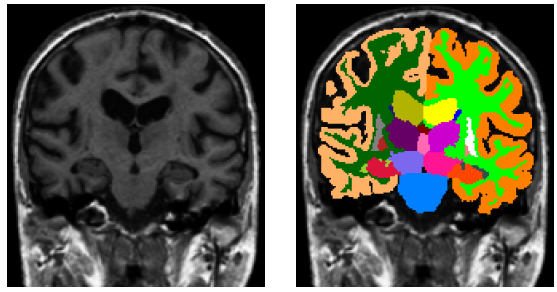


**Figure 3.6:** On the left an example slice from the cross-scanner dataset and on the right a corresponding manual segmentation.

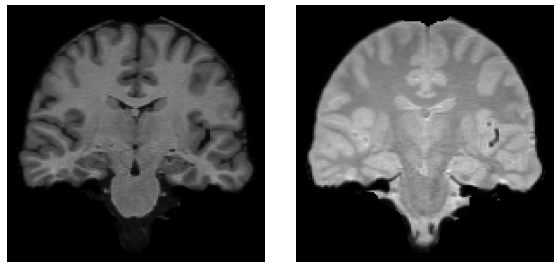


**Figure 3.7:** An example of the T1- (flip angle = 30°) and PD-weighted (flip angle = 3°) scans of the same subject from the multi-echo dataset.

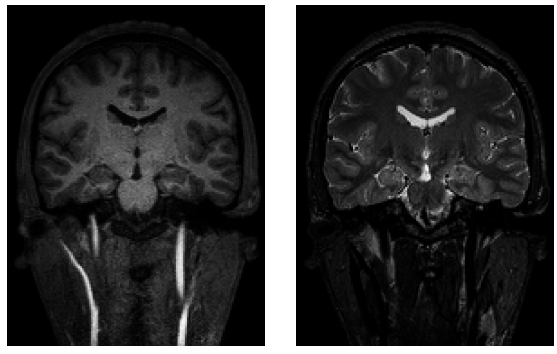


**Figure 3.8:** An example of the T1- and T2-weighted scans of the same subject from the test-retest dataset.

proach to four benchmark methods, then we evaluated the effect of the training set size on segmentation performance, next we investigated the multi-contrast segmentation performance of our method and finally we experimented with extending our framework to support multiple atlases.

**Experiment 1.** In the first experiment, we compared[3] our method against four state-of-the-art segmentation methods on the intra- and cross-scanner data sets. The benchmark methods are: **BrainFuse** [SYVL+10], **PICSL MALF** [WSD+13], **FreeSurfer** [FSB+02] and **Majority Voting** [RBMMJ04, HHA+06], which were previously described in chapter 2. As mentioned, FreeSurfer represents a supervised parametric segmentation method whereas BrainFuse, PICSL MALF and majority voting are non-parametric methods. The main interest of this experiment is two-fold: first, as all the methods were trained on the same data set, it enables us to compare the segmentation performance of our approach to the best methods in the field on the intra-scanner data. Second, we are interested in how the performance of the supervised methods is affected when applied to data coming from a different scanner. Furthermore, the parameter settings of each method were tested and tuned on the training data and each method was applied in exactly the same manner to both the intra- and cross-scanner data, which allows us to evaluate how robust these methods are "out-of-the-box" for different data sets.

For computing the pair-wise registrations in the multi-atlas approaches, we used the diffeomorphic ANTs/SyN framework [AEGG08] for PICSL MALF and majority voting, whereas diffeomorphic Demons were used for BrainFuse [SYVL+10]. These choices were based on which registration method was used in the original publications [WSD+13, SYVL+10].

Finally, although the manual labeling approach includes 39 different brain structures, we use a relevant subset of these structures listed in table 3.1 for validation. These structures were chosen as they are used for validation in other studies [FSB+02, SYVL+10], therefore making comparison with different studies easy.

**Experiment 2.** In the second experiment, we evaluated the computational efficiency of the different methods. The running times were measured on a cluster where each node has two quad-core Xeon 5472 3.0GHz CPUs and 32GB of RAM. We only used one core in the experiments in order to make fair compar-

[3]Comparison is done using Dice scores defined as: $\text{Dice} = 2|\mathbf{l}_A \cap \mathbf{l}_M|/(|\mathbf{l}_A| + |\mathbf{l}_M|)$, where $\mathbf{l}_A$ and $\mathbf{l}_M$ are the automatic and manual segmentations respectively and $|\cdot|$ is the cardinality of a set.

| Brain structures | Acronym |
|---|---|
| Left/Right hemisphere Cerebral White Matter | WM |
| Left/Right hemisphere Cerebellum White Matter | CWM |
| Left/Right hemisphere Cerebral Cortex | CT |
| Left/Right hemisphere Cerebellum Cortex | CCT |
| Left/Right hemisphere Lateral Ventricle | LV |
| Left/Right hemisphere Hippocampus | HP |
| Left/Right hemisphere Thalamus | TH |
| Left/Right hemisphere Putamen | PU |
| Left/Right hemisphere Pallidum | PA |
| Left/Right hemisphere Caudate | CA |
| Left/Right hemisphere Amygdala | AM |
| Brain Stem | BS |

**Table 3.1:** List of structures the segmentation performance is compared on.

isons, even though all the algorithms can potentially be parallelized. However, we also recorded the execution time of a multi-threaded implementation of our method, using 8 cores on a computer with 8 dual-cores with 3.4Ghz CPU and 64GB of RAM. This was done in order to enable us to compare the running time of our algorithm with those reported by other studies in the literature.

**Experiment 3.** In the third experiment, we studied the effect of the number of training subjects on the segmentation performance. To achieve accurate segmentations, a representative training set is needed to capture all the structural variation one might see within the subjects to be segmented [AHH$^+$09]. However, some algorithms require less training data than others to approach their asymptotic performance, which represents a saving in manual labeling effort. We therefore randomly picked 5 sets of 5, 10 and 15 subjects from the training data, and re-evaluated the segmentation performance of the proposed method, BrainFuse, PICSL MALF and majority voting on the intra- and cross-scanner datasets.

**Experiment 4.** In the fourth experiment, we evaluated the ability of the proposed algorithm to segment multi-contrast MR scans in both the multi-echo and the test-retest data sets. Given a training set which only consists of T1-weighted scans and corresponding manual segmentations, using multi-contrast information is out of reach for all the methods we compared against in the first experiment. This is due to either their non-parametric modeling approach (BrainFuse, PICSL MALF, majority voting) or the supervised intensity model (FreeSurfer). To quantify the effect of using multi-contrast information, we first

ran our method using only one of the available scans and then using two scans with different contrasts. For the multi-echo dataset we first used only the T1-weighted images (i.e., flip angle 30°), and then both the T1- and PD-weighted (flip angle 3°) images. The resulting segmentations were then compared to the manual segmentations using Dice scores.

In a similar fashion we first segmented the two time points in the test-retest dataset using only the T1-weighted images, and then using both T1- and T2-weighted images. Because manual segmentations were not available for this dataset, we used absolute symmetrized percent change (ASPC) [RSRF12a] to quantify the differences in the automatic segmentations between the two time points. This metric is defined as the absolute value of the difference in volume, normalized by the mean volume:

$$\mathrm{ASPC} = \frac{2|V_2 - V_1|}{V_1 + V_2},$$

where $V_1, V_2$ are the volumes at the two time points. Ideally this number should be small, as the subjects were all healthy and the time between the scans was not so long.

**Experiment 5.** Inspired by the success of the multi-atlas approaches in brain segmentation [SYVL$^+$10, WSD$^+$13, LW12], we were interested if extending our framework to support multiple atlases would be beneficial. The derivations of how the single-atlas framework can be extended to support more than one atlas can be found in Appendix A. In short, the generative model now assumes that each voxel $i$ is generated from one of $M$ atlases. Once we know which atlas generated the voxel, we draw the label $l_i$ from that atlas. The MR data is then generated, as before, from the GMM assigned to label $l_i$.

To test the effect of using multiple atlases, we chose one of the 15 subject atlases used in experiment 3, and constructed three atlases of 5 subjects and five atlases of 3 subjects by randomly dividing the 15 subject set to sets of five and three respectively. We then compared the segmentation performance of the multi-atlas set-ups to the single-atlas one on the intra-scanner data set. This was done using three different initialization schemes:

- First, we "pre-registered" each of the multiple atlases to the target scan independently following the single-atlas segmentation approach. The learned mesh node positions $\{\hat{\mathbf{x}}^m\}$ were then used as an initialization to the multi-atlas version of the algorithm, where we only updated the likelihood model parameters keeping the mesh node positions fixed. This is comparable to the approach most multi-atlas methods take, where the

| | Intra-scanner data | Cross-scanner data |
|---|---|---|
| Method | Average Accuracy | Average Accuracy |
| Proposed | 0.863 | **0.807** |
| BrainFuse | 0.868 | 0.744 |
| PICSL MALF | **0.896** | 0.760 |
| FreeSurfer | 0.853 | 0.799 |
| Majority Voting | 0.883 | 0.698 |

**Table 3.2:** Mean Dice scores of the different methods over the structures listed in table 3.1 for the intra-scanner (first column) and cross-scanner (second column) datasets.

pair-wise registrations are first computed and then followed by a label fusion step [SYVL$^+$10, WSD$^+$13, HHA$^+$06].

- Next we used the same initialization procedure, but also updated the mesh node positions allowing for further registration of the atlases.

- Finally, we ran the algorithm with no initial registrations, and optimized over the full parameter set consisting of the likelihood model parameters and the mesh node positions of each atlas.

### 3.3.3 Results: a short summary of the main findings

#### 3.3.3.1 Experiment 1: Intra-scanner and cross-scanner segmentation performance

The Dice scores between the manual and automated segmentations for each method on the intra- and cross-scanner data sets are shown in fig. 3.9. Table 3.2 collects the average score over the structures for each method and data set.

The results show that all of the methods perform well on the intra-scanner data set, which was expected as the properties of this data set are identical to the training data. The highest average score is achieved by PICSL MALF. However, on the cross-scanner data set, where the target scan properties are different from the training data set our approach achieved the highest mean score.

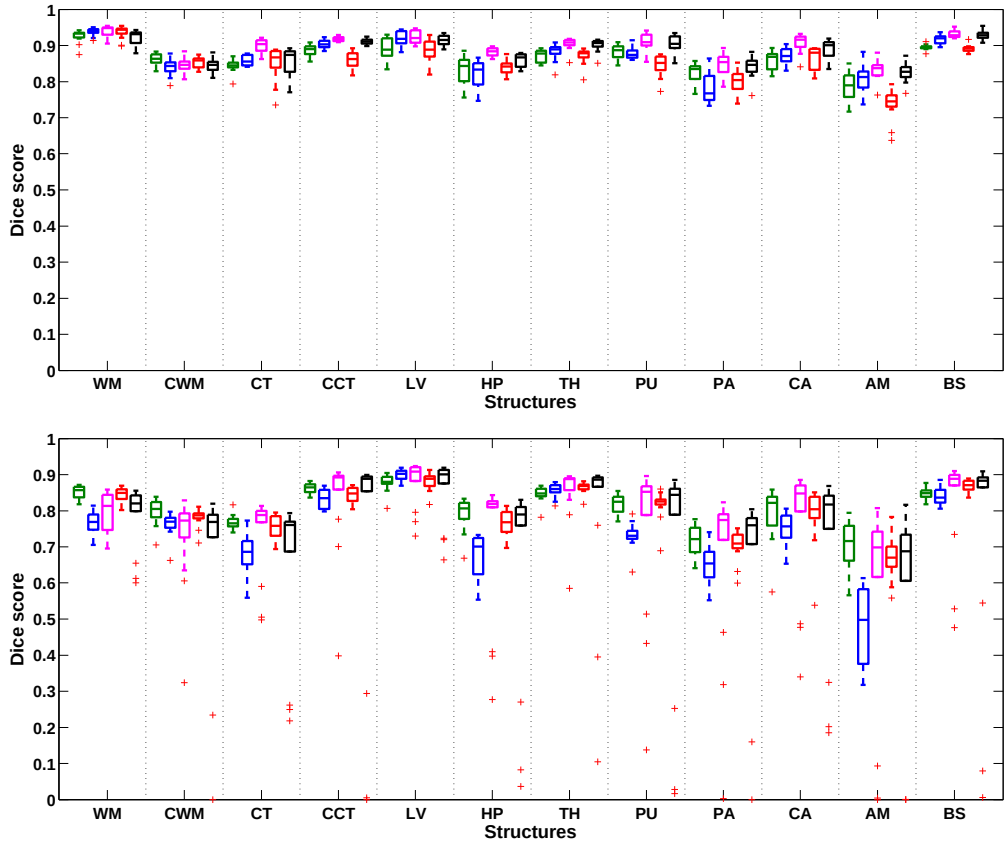



**Figure 3.9:** The Dice scores of the different methods for the intra-scanner (top) and cross-scanner (bottom) data. The proposed method = green, BrainFuse = blue, PICSL MALF = magenta, FreeSurfer = red and Majority Voting=black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'. For left and right hemisphere structures the scores are averaged over the hemispheres. See table 3.1 for the acronyms.

| | Average time per subject (h) | | |
|---|---|---|---|
| Method | Reg. | Fusion | Full Time |
| BrainFuse | 16 | 1 | 17 |
| Majority voting | 143.9 | 0.1 | 144 |
| PICSL MALF | 143.9 | 3.8 | 147.7 |
| FreeSurfer | - | - | 9.5 |
| Proposed | - | - | **1.4** |

**Table 3.3:** Mean computational time for the different methods. For label fusion methods the computation times for registration (Reg.) and label fusion (Fusion) are listed separately.

#### 3.3.3.2 Experiment 2: Execution time

The approximate mean computation time for a single scan using the different methods is shown in Table 3.3. The results show that the proposed method is approximately 7 times faster than FreeSurfer, 12 times faster than BrainFuse and 100 times faster than PICSL MALF and majority voting.

Here the main finding is that the parametric methods (i.e., FreeSurfer and the proposed method) are significantly faster than the multi-atlas approaches. As mentioned in chapter 2, this is due to the multiple pair-wise registrations used in the non-parametric approaches, whereas in the parametric methods only a single non-linear registration between the atlas and the target scan is needed. Note, however, that even the fusion step in PICSL MALF is quite time consuming. The reason why our method is faster than FreeSurfer, is that instead of a standard voxel-based probabilistic atlas, our method employs a mesh-based atlas which is much sparser.

In the multi-threaded set-up, the proposed method has an execution time of 23.5 minutes per scan on average. The fastest whole-brain method to our knowledge is presented in [ZGC14], which builds on a random forest classifier, with execution times in the range of 5 to 13 minutes. However, due to the supervised approach this method does not handle contrast differences between training and target scans.

#### 3.3.3.3 Experiment 3: Effect of the number of training subjects

Figure 3.10 shows the performance of each method when trained on different number of training subjects for the intra- and cross-scanner data sets. The line is drawn through the average Dice score of each set and the bars show the

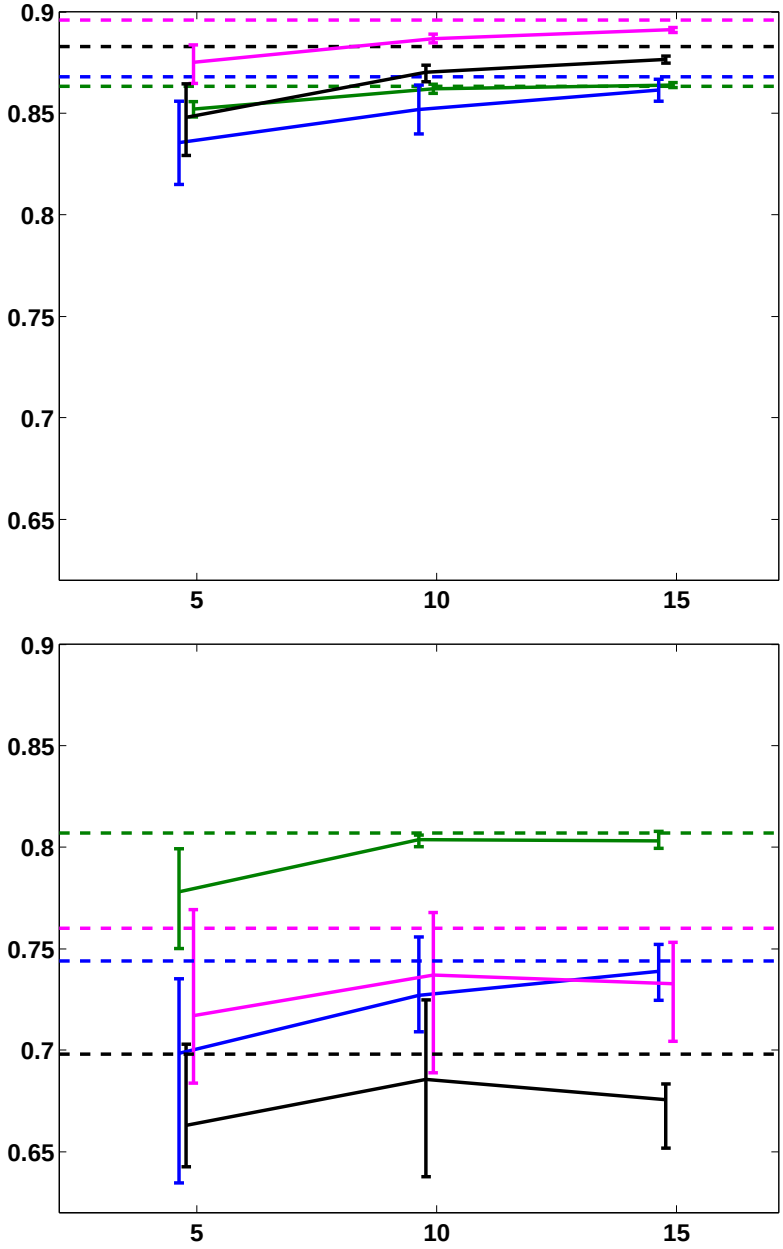



**Figure 3.10:** Average Dice scores for different number of training subjects for the intra-scanner (top) and the cross-scanner (bottom) data, as well as their variance across randomly selected sets of training subjects. The proposed method in green, BrainFuse in blue, PICSL MALF in magenta and majority voting in black. The error bars correspond to the lowest and highest average score for the random subset of subjects. The dashed line marks the Dice score obtained when all subjects in the training pool are used.

variance around the mean score.

The results show that adding more training subjects generally yields more accurate segmentations for all methods. However, the proposed approach reaches its maximum performance faster than the multi-atlas approaches. Also the variance of the mean score is small for all training set sizes, indicating that the performance of the proposed method does not depend much on the specific subjects included in the training set. The performance of the multi-atlas methods is more dependent on the number of training subjects especially on the cross-scanner data set, where the pairwise registrations are more challenging due to the different properties of the training and target data.

#### 3.3.3.4 Experiment 4: Multi-contrast performance

The Dice scores for the multi-echo dataset, when using only T1-weighted scans and when using both T1- and PD-weighted scans, are shown in Figure 3.11. The results are very similar whether or not the PD-weighted scan is included, which indicates that the PD-weighted contrast does not add much useful information for structural segmentation of healthy brains. Example segmentations of the multi-echo dataset using uni- and multi-contrast scans are shown in Figure 3.12.

The volume differences between the two time points in the 39 subjects of the T1/T2 test-retest dataset are shown in Figure 3.13. In general, they are quite similar and small for both single- (only T1) and multi-contrast (both T1 and T2) segmentations, with the median ASPC in the 1-2% range. There are some larger differences – especially in the thalamus and pallidum – when using multi-contrast data. This appears to be mostly due to imaging artifacts in the T2-scans, an example of which is shown in Figure 3.14. We note that this data set also has the lowest resolution of all the datasets we tested the method on, and thus partial volume segmentation errors are most prominent on this data set.

#### 3.3.3.5 Experiment 5: Extending to multiple atlases

The results for the different initialization set-ups for the multi-atlas approach, are shown in fig. 3.15. Allowing only likelihood model updates with pre-registered atlases gives very similar scores compared to the single atlas approach when three atlases built from five subjects were used. However, for the case of five atlases each built using three subjects, the scores are significantly worse especially in sub-cortical structures. Allowing for further registration of the mesh nodes using the same initialization process yields similar results. The algorithm

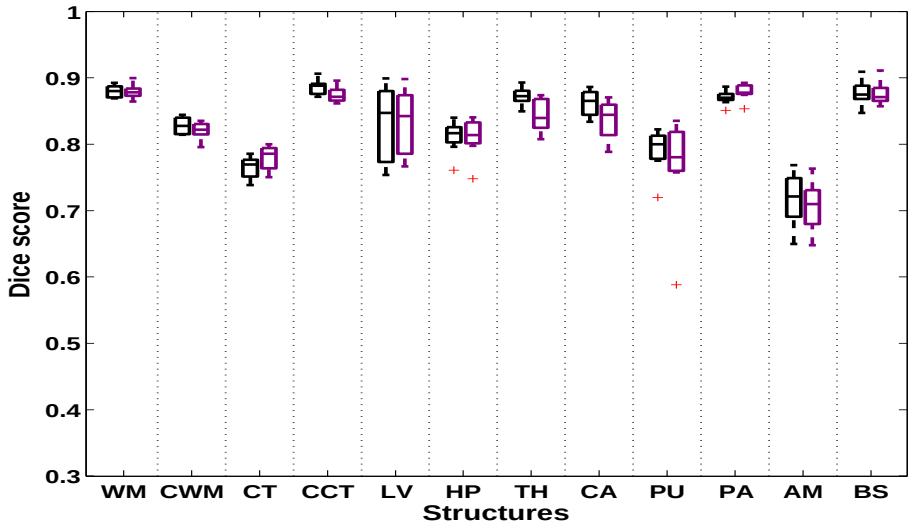



**Figure 3.11:** Dice scores for the multi-echo dataset. Performance on multi-contrast input data is shown in purple, and on T1-weighted data only in black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'.

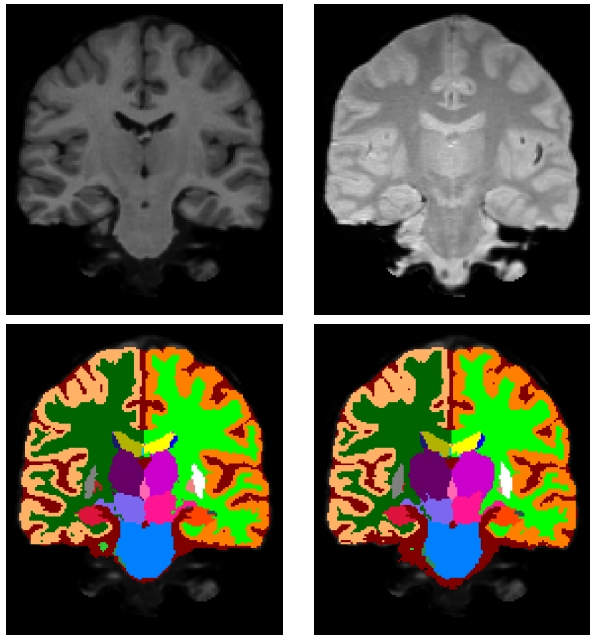


**Figure 3.12:** Top row: target scans, T1-weighted on the left and PD-weighted on the right. Bottom row: automatic segmentation using **only** the T1-weighted scan on the left, automatic segmentation using **both** scans on the right.

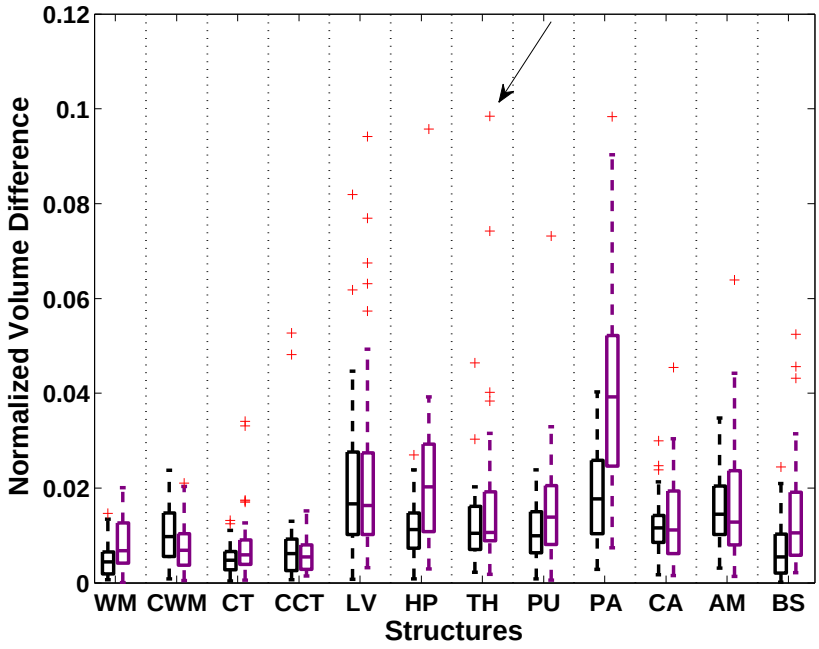



**Figure 3.13:** The ASPC scores for the test-retest dataset. Volume differences between the time points on multi-contrast input data is shown in purple, and on T1-weighted data only in black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'. The outlier marked by an arrow is the one shown in Figure 3.14.

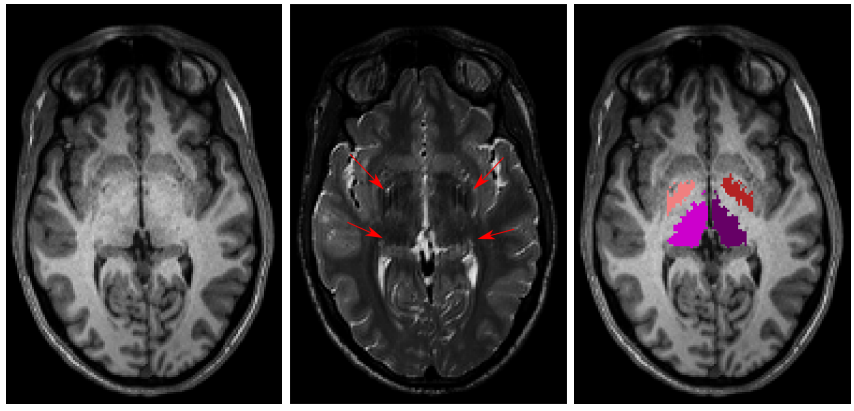


**Figure 3.14:** An example of an outlier subject marked by the arrow in Figure 3.13. From left to right: a T1-weighted scan with no visible artifacts, a T2-weighted scan with a line-like artifact in the pallidum and thalamus area marked by red arrows, and an automated segmentation of pallidum and thalamus showing the segmentation error caused by the artifact.

however only took on average three registration steps, which implies that the initial warp estimates were close to a local optimum. In the final set-up, when we did not use any initial registrations, but ran the algorithm from the starting positions of the meshes, the scores exhibit similar behaviour as in the previous set-ups.

## 3.4 Discussion

### 3.4.1 Experiment 1: Intra-scanner and cross-scanner segmentation performance

The results from the first experiment show the main up- and downsides of the different segmentation approaches discussed in chapter 2. On the intra-scanner data, where the properties of the training and target data are exactly matched, the multi-atlas approaches yield very good results and outperform our unsupervised generative segmentation framework. However, when applied to target data with different properties these methods can yield very varying results.

The principal error source for the multi-atlas methods is due to misregistrations between the training and target scans [WSD+13]. This explains the poor performance of the multi-atlas approaches on the cross-scanner data set. The registration framework employed in BrainFuse uses a sum-of-squared-differences (SSD) similarity measure, which is likely sub-optimal when applied on cross-scanner data. The registration framework employed for PICSL MALF and Majority Voting uses a different similarity measure based on the cross-correlation (CC) metric, which is more robust against intensity variations. However, there are still some subjects in the cross-scanner data set for which computing the registrations is very difficult as shown by the outliers in figure 3.9. Note that Majority Voting achieves very good scores on the intra-scanner data set, but does quite poorly on the cross-scanner data set. This shows that the performance of the multi-atlas methods is mainly driven by the accuracy of the registrations. If accurate registrations are available, there is no need to use elaborate fusion strategies. However, when this is not the case, weighted voting helps to downplay the effect of the poorly registered subjects.

As explained in chapter 2, FreeSurfer uses a supervised intensity-model where the Gaussian mixture parameters are learned from the training data. Based on this, one might expect to see a larger dip in the segmentation performance of FreeSurfer on the cross-scanner data. The relatively good performance is due to an in-built correction step which is performed for T1 acquisitions. The correction

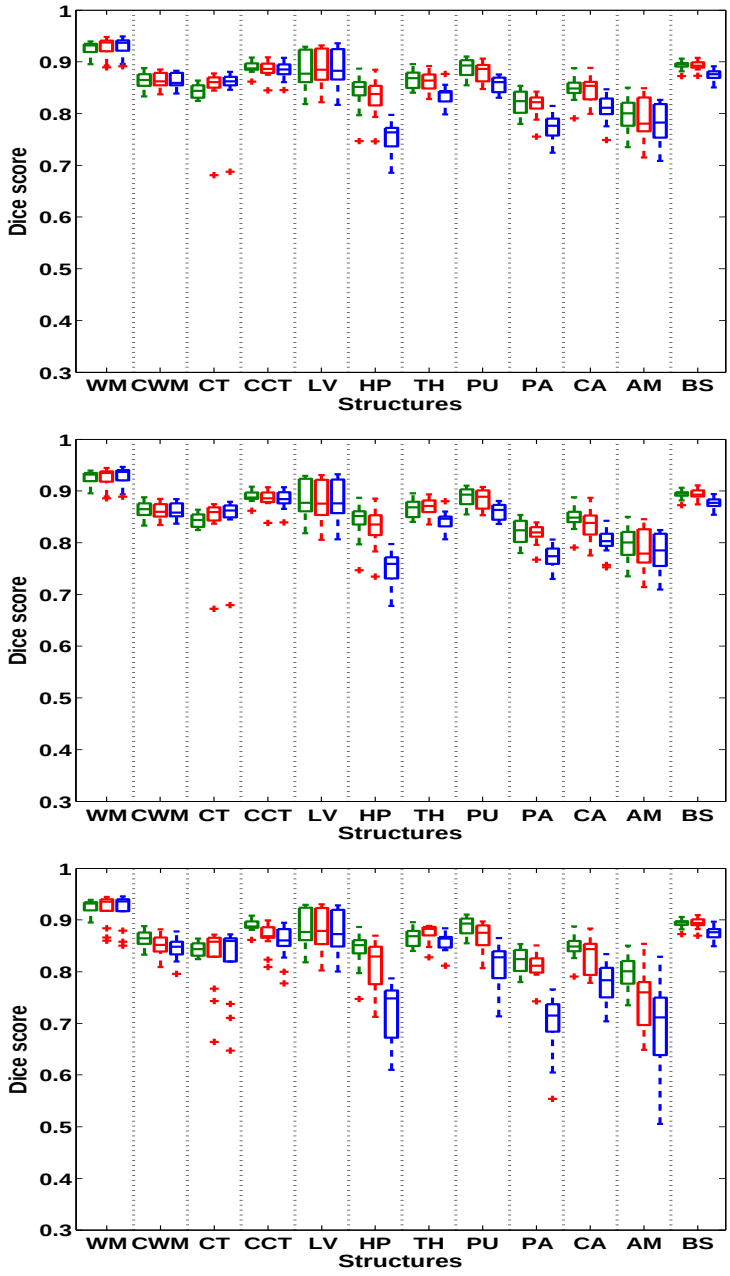



**Figure 3.15:** Dice scores of the label fusion approach: 15 subject atlas in green, three five subject atlases in red and five three subject atlases in blue. The top figure shows the scores using only fusion, the middle one with additional registration and the bottom one without any initializations. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'.

applies a multi-linear atlas-image registration and a histogram matching step to update the class-conditional prior intensity profiles for each structure to better match the intensity-properties of the structures in the input scan [HF07]. It is interesting to note that our approach actually outperforms FreeSurfer on both data sets, even though FreeSurfer explicitly encodes a priori information about the intensity properties of the different structures, whereas the proposed method assumes an uniform prior on the intensity parameters. It might be worthwhile to experiment with including this type of prior information into our framework in order to see if it has any effect on segmentation accuracy. However, this would naturally make the method less general, and thus was not attempted during this project.

All the benchmark methods we compared against require some post- and/or pre-processing steps to be performed. In the FreeSurfer pipeline the target scan is first skull-stripped, bias field corrected and intensity normalized before the segmentation is performed. Most of the multi-atlas methods, such as BrainFuse, have similar pre-processing steps. Furthermore, PICSL MALF refines the registrations using a local search after the registrations have been computed. These kind of sequential data processing pipelines can however lead to some difficulties: first, the performance of the applied method can become very dependent on the exact form of the pre-processing (or post-processing) as each step in the pipeline relies on the previous ones. As an example, a failure in skull-stripping will most likely have a negative impact on the resulting segmentations, as there is no mechanism in the pipeline to correct for the failed initialization. Second, objective comparison between different segmentation approaches can become difficult if different pre-processing steps are used. We can not say for sure if a certain approach does well because of the modeling framework or because the pre-processing was done well. Finally, the pre-processing steps themselves often consist of quite elaborate modeling frameworks. For example, the widely used N3 bias correction algorithm [SZE98] can be interpreted as a generative probabilistic model [LIVL14], whereas the ROBEX skull stripping tool [ILTT11] uses a combination of generative and discriminative models to segment the brain from the non-brain structures. Thus, the sequential pipeline often consists of many separate steps which already perform segmentation in one form or the other. In this project, we have tried to build a unified model in which all the needed steps are performed in a joint manner. We emphasize that, due to this approach, the proposed method does not require any special pre- or post-processing steps but performs robustly on different types of data sets out-of-the-box.

The main take-home-message from this experiment is that supervised non-parametric methods can yield highly accurate segmentations when the training and target data are identical, but when they are applied across scanner platforms care must be taken in order to get good results. First, the similarity metric used in the registrations should be robust to intensity differences between the scans.

Second, heavy pre- and post-processing of the training and target scans is likely needed but can make the methods very dependent on the success of these steps. Finally, weighted-voting is necessary to downplay the effect of poorly registered scans. We note that re-tuning the methods, including pre-processing steps, on the cross-scanner data would likely be beneficial for the performance, but the aim of the experiment was to see how well these methods work when applied in a scenario where perfectly matched training data is not available for all the scans to be processed. Although the segmentation accuracy of our approach slightly trailed that of the very best multi-atlas techniques on the intra-scanner task, the proposed method performs robustly on data coming from different scanner platforms without any need for re-tuning and pre- or post-processing. There are many situations, apart from clinical applications which we have discussed before, where this robustness to intensity changes in the input scans is beneficial: for example in the study of the hippocampus T2-weighted scans are often used [ISV13b], whereas when studying the developing brain the white matter myelination process leads to a complete reversal of the MR signal in the course of the first year after birth [PCE+01]. In such cases, we can not apply the supervised approaches without acquiring extra training data with the same scan protocol.

### 3.4.2 Experiment 2: Execution time

As discussed above, accurate registrations are required in order to obtain accurate segmentations using multi-atlas approaches. These registrations are however costly to compute. The results show, as expected, that the parametric approaches yield significantly faster computational times compared to the multi-atlas approaches. The ANTs/SyN registration framework used for PICSL MALF and majority voting is especially time consuming taking almost 6 days to compute the 39 pair-wise registrations from the training data to a single target scan. Compared to the time taken for registration, the fusion step is fairly fast, although in PICSL MALF this step is also quite computationally demanding due to the local search that is applied to refine the registrations.

The proposed method is significantly faster than all the other benchmark methods. As explained before the difference in execution time compared to FreeSurfer, which also is a parametric method, is mainly due to the sparse encoding of the mesh prior, and also because our modeling approach obviates the need for any pre- or post-processing.

The main take-home-message from this experiment is that, the good performance of the non-parametric methods comes with a high computational cost. Although computational power is fairly cheap today, the amount of data to

| Number of subjects | Average number of vertices |
|---|---|
| 5 | 33,606 |
| 10 | 44,614 |
| 15 | 51,258 |

**Table 3.4:** Average number of vertices in the proposed atlas mesh for different numbers of training subjects.

process is also growing fast. The ability to obtain segmentations fast facilitates experimenting in large studies, and is essential in clinical flow where many patients are scanned daily. We note that the competing methods can of course be parallelized, but even on the 8 core computer used for speed-testing the proposed method, the theoretical limit for the execution time of PICSL MALF would be roughly 19 hours, whereas our approach achieved an execution time of 24 minutes.

### 3.4.3 Experiment 3: Effect of the number of training subjects

The results from the third experiment show that larger training sets generally yield increased performance for all methods. However, the proposed method approaches its maximum performance faster than the multi-atlas methods. Even with only five training subjects the segmentation accuracy of the proposed method is already good, with mean accuracy 98.5% of the maximal performance on the intra-scanner dataset and 96% of the maximal performance on the cross-scanner dataset. The variance around the mean score is also small, which indicates that the performance of the method is not dependent on which specific subjects are included in the training set. This robustness towards the training set size is likely due to the atlas construction process. As explained in section 3.1.1, the topology of the mesh is optimized given the training labelings. Thus, the optimal amount of smoothing is automatically estimated when learning the atlas parameters. This effect is shown in table 3.4, which lists the average number of mesh vertices for the atlases constructed from the 5, 10 and 15 subject training sets. The atlases constructed using only 5 subjects are significantly sparser compared to the 10 and 15 subject groups, thus yielding probability-wise "smoother" atlases which are not so prone to over-fitting.

For the multi-atlas methods the performance is more dependent on the number of available training subjects, especially for the cross-scanner dataset. On the intra-scanner dataset, PICSL MALF achieves a good mean score already with 5 subjects, but the performance increases more slowly compared to the proposed

method. The variance of the score is also larger, especially for the 5-subject set, showing that the performance is dependent on the particular subjects included in the training set. The same behaviour is observed for majority voting and BrainFuse, but with larger variances over all the subjects sets. On the cross-scanner dataset the performance of all multi-atlas methods varies significantly even when trained on 15 subjects, which is due to the registration difficulties between the training and target scans. In [AHH+09] the authors suggest ways of making the performance of multi-atlas methods more robust by pre-selecting a group of training subjects that are most similar to the target scan, which would be particularly beneficial for majority voting. However, this technique assumes that there is a large training set of many different subjects, from which we can do this selection from. As discussed before, obtaining large training sets can be quite costly due to the tedious manual segmentation process. Thus, being able to perform well using smaller data sets represents savings in manual labeling effort.

The main take-home-message from this experiment is that multi-atlas methods seem to benefit from larger training sets, whereas the proposed approach appears to have robust and repeatable performance even with small training sets. This can be advantageous for certain populations, such as infants, where large amounts of training data is difficult to obtain. Furthermore, robustness to different training subjects is also an advantage, as it allows the whole training set to be used for learning without the need to discard training subjects that might be problematic.

### 3.4.4 Experiment 4: Multi-contrast segmentation performance

One main benefit of our generative parametric segmentation approach is that it readily extends to any number of input contrasts. Although the results indicate that using multi-contrast information does not increase the segmentation accuracy for healthy subjects, this multi-contrast information is *necessary* when we aim to detect pathologies [GLFN+13], as the T1-weighted contrast alone can not be used for locating lesions or tumours.

One problem with including extra contrasts for structural segmentation is that sometimes they include information that is not helpful. This is highlighted in the results for the test-retest data set where artifacts only present in the T2-weighted scans negatively impact the segmentations. As mentioned earlier this data set also has a fairly low resolution, thus some of the volume differences are likely due to partial volume effects, which could be helped by modeling them explicitly as in [VLMVS03]. Furthermore, the two time points were processed completely

independently and the results would likely be better if a dedicated longitudinal segmentation model was used [RSRF12b]. Development and experimentation with such a model is left for future work.

On the multi-echo data set, the results for the uni- and multi-contrast set-ups are very similar, although it looks like some of the main tissue classes can be better separated when using multi-contrast data as evidenced by the slightly higher score for the cerebral cortex. The similarity in the performance is most likely due to the fact that the PD-weighted scan does not include too much extra information, especially for separating the sub-cortical structures of the brain. One more subtle problem in validating the performance in multi-contrast data sets against manual segmentations is that the expert radiologists typically base their segmentations only on T1-weighted contrast, which is also the case for this data set [ISV13b]. Using information from both contrasts in the manual segmentation process might yield different structure boundaries, but, as can be seen from figure 3.12, the T1-weighted scan typically offers the best contrast for outlining the structures.

The main take home message from this experiment is that multi-contrast information might not be beneficial for the purpose of structural segmentation in healthy brains but, as we will see in chapter 4, is needed when pathologies are present.

### 3.4.5 Experiment 5: Extending to multiple atlases

The results of the multi-atlas experiment show that substituting an atlas built from 15 training subjects with multiple atlases built from subsets of the training set does not result in any significant increase in segmentation accuracy. There are two main reasons for this: first, the probabilistic atlases built from a subset are much sparser and thus much smoother probability-wise, as discussed in experiment 3. This is likely not beneficial especially in sub-cortical structures, which typically exhibit low contrast i.e., borders between the structures are hard to detect. Here the shape and location information encoded by the prior becomes much more important as the structures can not be separated based on intensity information. The results support this explanation, as the sub-cortical scores are consistently lower, especially when using 5 atlases of three subjects, for the multi-atlas approach. The second reason is due to optimization difficulties, as the number of parameters to optimize increases significantly with the number of atlases. From table 3.4, we see that the 15 subject atlas has on average around 44000 nodes whereas a five subject atlas has around 33000 nodes. Thus for three five subject atlases we need to optimize the positions of almost 100000 nodes, which is likely very prone to local minima and also makes convergence

very slow. Furthermore, the node positions of different atlases are intertwined, as seen from the update equations in Appendix A, which further complicates the problem. This is evidenced by the results of the final experiment where no initialization was used, which resulted in a noticeable reduction in segmentation accuracies in some of the sub-cortical structures when 5 atlases of three subjects are used.

A somewhat similar approach combining EM and multi-atlas labeling has been suggested in [LHA+12], where the authors first warp the training images to the target scan and then create a probabilistic atlas from the warped segmentations using locally weighted fusion. Based on this atlas a GMM is fitted to the target image intensities using EM, and further regularization is enforced using an MRF prior. The pair-wise registrations are computed using the so-called MAPER [HKL+12] framework, which incorporates tissue probability maps into the registration process. Thus the whole pipeline consists of many sequential segmentation and registration steps, which are inherently circular. In the proposed multi-atlas framework the atlases are jointly registered to the target image and label fusion is naturally included into the likelihood model assuming that the neuroanatomical label in each voxel is generated from a single atlas. We hypothesised that this approach would be beneficial, but due to the problems discussed above this was found not to be true. Furthermore, the multi-atlas approach has the downside of being computationally quite demanding, and as one of the main aims of this project was to devise a method with a low computational cost, it seems that a single atlas approach is the way to go.

Chapter 4

# Generative modeling for joint whole-brain and lesion segmentation

In this chapter we extend the whole-brain segmentation model of the previous chapter to include a model of lesions. The chapter is constructed as follows:

- First, we motivate the problem, and review some of the approaches that have been used for lesion segmentation.
- Next, we explain how the general modeling framework is extended to include a model for lesions.
- Then, we introduce our lesion shape model, which is based on convolutional restricted Boltzmann machines, and show how it is incorporated into the model.
- After the full model is in place, we show how new target MR scans with lesions can be segmented using the model.
- Finally, we overview the experiments and results from paper C, and conclude with a discussion.

## 4.1 Introduction

Multiple sclerosis (MS) is the most common inflammatory disorder of the central nervous system, and the leading cause of non-traumatic neurologic disability in young adults in the US and Europe [MRV13]. In the US, multiple sclerosis is the second most costly chronic health condition after congestive heart failure [MRV13]. MS results in demyelination of white matter tracts in the brain and is characterized by the formation of lesions within the white matter [MK10]. These white matter lesions are frequently associated with motor disorders, dizziness, depression and a variety of other physical and neurological symptoms [MK10]. An example multi-contrast MR scan of a patient suffering from MS disease is shown in fig. 4.1, note the lesions which are highlighted in the FLAIR contrast.

The severity of MS has been shown to correlate with the mean annual treatment costs of an individual patient, going from 20000€ for a mild type to around 80000€ for the most severe type [KBJ06]. Thus, in order to relief the societal burden, and to increase the quality of life for patients, it is crucial to be able to diagnose the disease early and track it accurately to assess treatment efficacy. Currently diagnosis and tracking is done by experts based on a neurological examination and visual inspection of MR scans of the brain. The visual inspection is often combined with manual lesion segmentations to aid the disease assessment. This can however be problematic, because the expert lesion delineations have been shown to have large inter- and intra-expert variability [GLFN$^+$13, LOC$^+$12] and are very time consuming to obtain. Thus, robust automatic tools for segmenting the lesions in MR images would greatly help in disease diagnosis and patient follow-up.

### 4.1.1 Current and previous approaches to lesion segmentation

Here we overview some of the many approaches that have been suggested for lesion segmentation. These methods can be, following the definitions in chapter 2, roughly divided to supervised and unsupervised approaches [GLFN$^+$13]. For a more thorough review with performance comparisons, the reader is referred to the work in [GLFN$^+$13, LOC$^+$12].

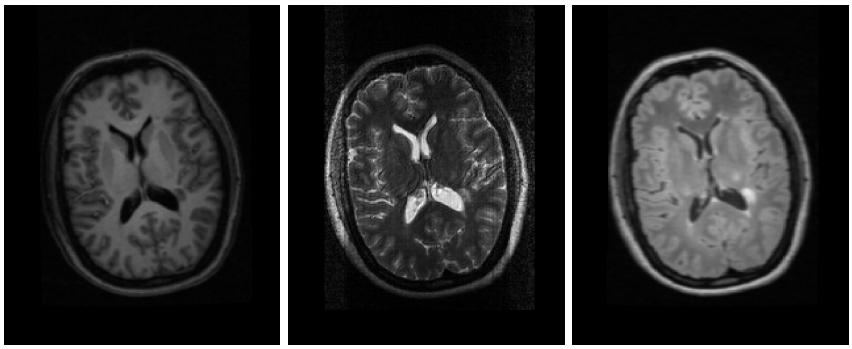


**Figure 4.1:** An example multi-contrast MR scan of a patient with lesions. From left to right: T1-weighted scan, T2-weighted scan and a FLAIR scan. The lesions are especially well visualized in the FLAIR scan as bright outliers around the ventricles.

#### 4.1.1.1 Unsupervised approaches

The earlier lesion segmentation methods often built upon the generative models used in tissue classification. These methods typically either modeled lesions as a separate structure class, or as outliers to the normal healthy[1] tissues. One of the earliest approaches [KGM+99], extended the tissue classification method presented in [WIG+96] by including an extra Gaussian distribution to model the lesions. After the initial segmentation to the three tissue classes and lesions, post-processing steps based on morphological operations and size of the lesion clusters were applied to remove outliers that were mainly due to partial volume effects [KGM+99]. In [DPGL+04] a similar approach is used, but extra Gaussian distributions are added in order to model the partial volume effects. Here, the lesions were not initially modeled by any of the Gaussians, but were extracted after the initial segmentation by first detecting outliers based on the Mahalanobis distance, and finally fitting a mixture of two Gaussians to this data which then allowed for separating the true lesions from other outliers. This method was further extended to include morphological post-processing of the lesion segmentations in [SLAM08], which was the winning method in the 2008 MICCAI challenge on lesion segmentation [SLC+08]. Because lesions can have very varied intensity properties even within a patient, the authors in [FGG09] model each tissue class and lesions with a large number of spatially constrained Gaussians. Each of the Gaussian distributions then models the intensities of a local spatial area allowing for a more complex model of intensities. In [FGG09]

[1] Here healthy refers to non-lesioned tissue, but in reality the surrounding structures are also affected by the disease.

a level-set based post-processing is used to refine the lesion borders.

Approaches that do not try to model lesions explicitly typically try to built outlier-robust models of the tissue classes and then detect lesions as observations that do not fit the model. The method presented in [VMVS01] builds on an atlas-based tissue segmentation method presented in [VMVS99b], but uses an M-estimator approach [VMVS01] in order to make the model robust to outliers. The lesions are then detected as data points that have a low probability of being generated from the model based on the Mahalanobis distance. Further regularization is enforced by adding restrictions to the intensities of the expected lesions and using an MRF-prior to encourage clustered lesion segmentations. Similar methods are presented in [PG08] and [GLPA$^+$11], where in [PG08] a minimum covariance determinant is used for robust estimation, whereas in [GLPA$^+$11] the authors rely on a trimmed likelihood estimator. However, both methods rely on Mahalanobis distance thresholding to identify the lesions.

#### 4.1.1.2 Supervised approaches

Recently, more and more methods build upon supervised classifier-based approaches. These methods can be summarized, somewhat simplistically, into two stages: generating suitable image features for lesion detection and selecting a suitable classifier to be trained with these features. Once such a classifier has been trained on a set of MR scans and manual lesion segmentations, lesions in a target scan can be easily detected by pushing the data through the classifier. Early approaches typically used only a handful of features for training: in [AVV08] the authors train a k-nearest-neighbour (KNN) classifier using intensity values and spatial location of voxels as features. The voxels in the target scan are then classified by assigning each voxel the label of the nearest feature cluster. A similar approach is presented in [ZFE02], where a three-layer artificial neural network (ANN) is trained using three intensity features from T1-, T2- and PD-weighted images along with the a priori tissue probabilities for each voxel obtained from a probabilistic tissue atlas. Instead of using only a few features, in [MTTT08, WHH08] the authors create thousands of features, based on pre-defined filters, computed from patches placed around each voxel. The features are then used to learn an ensemble of weak AdaBoost classifiers which are combined using a probabilistic boosting tree. However, as pointed out in [GCM$^+$11], most of these methods only consider *local* features such as intensities or tissue probabilities in a voxel, or features derived from small patches placed around each voxel. Currently the most successful classifier-based methods use extended spatial neighborhoods to provide rich contextual information for increased segmentation accuracy. In [GCM$^+$11] this is achieved by training a random forest (RF) classifier using a combination of local and context-rich

features. In [KRA+13] the authors use a conditional random field (CRF) framework incorporating multi-level features to detect gadolinium enhanced lesions. On the first level candidate lesions are detected using an RF classifier based on local intensity and spatial features. On the second level, higher-order textural patterns computed in a bounding box around the candidate lesions are used to refine the segmentations.

### 4.1.2 Limitations of the existing approaches

In both of the review articles [GLFN+13] and [LOC+12] the authors conclude that automatic lesion segmentation still remains an open problem. In [GLFN+13] one of the main issues brought forth is how to handle multi-center data sets. This relates to the arbitrary intensity levels of the MR scans, discussed also in chapter 2, where scans from different scanners can have different intensity properties even when the same imaging sequence is used [GLFN+13]. Another problem is that although MR is the central tool to study MS disease due to its ability to visualize the lesions, there still exists no standardized clinical MR protocol to study white matter lesions [FRA+06]. Many different sequences, such as T2-weighted, PD-weighted or T2-FLAIR, can be used for highlighting the lesions in the brain. This can be problematic when using a supervised approach, which assumes that the training and target data have similar intensity-properties. Furthermore, as discussed in relation to experiment 1 in the previous chapter, to ensure robust performance the supervised approaches typically require a lot of pre-processing steps such as brain extraction, bias field correction and intensity normalization, which have a large effect on the quality of the resulting segmentations. The unsupervised approaches are readily sequence-adaptive, and typically also include explicit models of the bias fields. However, given that the pre-processing is done carefully, the supervised methods have shown to outperform the unsupervised approaches [GCM+11].

Another fundamental problem hampering the application of the current tools for the study of MS disease, is that most of them only provide segmentations of lesions. This is especially true for the classifier-based approaches which are usually trained to segment the target scans into either lesion or background. However, even if accurate segmentations of lesions are available, the biomarkers derived from them, such as number of lesions or full lesion volume, have been shown to correlate poorly with clinical disability per se [JNS+13, GMLB+14]. This is possibly due to an underestimation of the importance of regional brain atrophy in diseases like MS [JNS+13, GMLB+14, FRA+06]. Finally, while most unsupervised approaches can segment lesions and tissues, having segmentations of the different neuroanatomical structures would possibly provide a richer set of biomarkers. This might facilitate better understanding of the disease mech-

anisms behind MS [GMLB+14].

In the next section we extend the whole-brain segmentation framework from the previous chapter to include a model of lesions. As a lesion model we use a convolutional restricted Boltzmann machine (cRBM) [Smo86, LGRN09, NRM09], which provides much richer spatial models compared to the low-order MRFs that have traditionally been used in the field for spatial regularization of lesion segmentations.

As pointed out in [GLFN+13] it is necessary to include spatial information on two different levels in order to obtain accurate lesion segmentations. The importance of doing this has been further emphasized by the success of the classifiers using local and context-rich features. The first level is within a local neighborhood, in order to reduce the impact of noise related to the voxel intensities and to increase the coherence of the segmentations. The second level is anatomical, in order to specify typical areas for lesion occurrence.

As we will see, the proposed framework takes into account both of these levels due to the incorporation of the detailed mesh-based probabilistic prior and the high-order neighborhood information encoded by the cRBM. Furthermore, the model is based on an unsupervised approach thus remaining agnostic to the number and contrast of input MR scans while obtaining segmentation of *both* lesions and the surrounding neuroanatomy. Simultaneous whole-brain and lesion segmentation has been attempted before [SBO+10], but the method we propose segments considerably more structures and learns spatial lesion models automatically from expert lesion segmentations as opposed to relying on a set of hand-crafted rules to remove false positive detections.

## 4.2 Generative model for joint whole-brain and lesion segmentation

In order to extend the generative model from chapter 3, we need to incorporate lesions into the prior and likelihood models. This is achieved by defining a *joint* segmentation prior, $p(\mathbf{l}, \mathbf{z})$, over the neuroanatomical labels $\mathbf{l}$ and lesions $\mathbf{z}$. In the same manner, the likelihood model will now become dependent of both $\mathbf{l}$ and $\mathbf{z}$: $p(\mathbf{D}|\mathbf{l}, \mathbf{z})$. A graphical presentation of the extended generative model is given in fig. 4.2.

Similar to chapter 3, we then look for segmentations that maximize the posterior

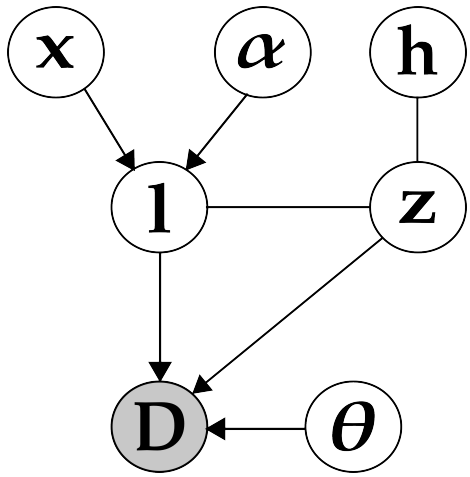



**Figure 4.2:** Graphical model of the generative segmentation framework extended to include lesions **z**. The variables **h** are so-called hidden units, which are further described in the next section. The target data **D** is observed which is denoted by the shading.

probability of **l** and **z** given the data **D**:

$$p(\mathbf{l}, \mathbf{z}|\mathbf{D}) = \frac{p(\mathbf{D}|\mathbf{l}, \mathbf{z})p(\mathbf{l}, \mathbf{z})}{p(\mathbf{D})}. \tag{4.1}$$

In the following we detail the exact form of the extended prior and likelihood models, but first we start with a description of the cRBM lesion model.

### 4.2.1 Generative lesion shape model using convolutional restricted Boltzmann machines

In order to give the reader some insight into our lesion model, we first introduce the basic restricted Boltzmann machine (RBM) [Smo86] model, which does not rely on convolutions, before we overview the convolutional variant. For convenience and clarity, the notation is presented for a 1D case, but all models readily extend to a 3D case.

The RBMs have recently gathered a lot of attention in the field of machine learning as the building blocks in the so-called deep learning approaches [HOT06, SH09], which are multi-layer unsupervised generative probabilistic models. Typically these multi-layer frameworks are used to extract high-level features automatically from a large collection of training images, obviating the need for

feature engineering, i.e., choosing suitable features manually. The learned features can then be used for image classification tasks given an annotated training set. However, the basic RBM model has been used effectively to learn distributions over binary data, such as hand-written digits [Hin02], which makes them very interesting for our purpose: learning a generative model of binary lesion maps.

The basic RBM is a parametrized generative probabilistic model, which can be interpreted as a particular type of Markov random field model with a specific two-layer structure [Smo86, FI12, Sal15]. The model consists of a binary input or observation layer, which in our case corresponds to the lesion map, and a binary hidden unit layer, where each unit effectively acts as a feature detector [FI12]. The features are encoded through weighted connections *between* the visual and hidden layers. Figure 4.3 provides a graphical representation of the model. Given a binary input vector of length $I$ denoted by $\mathbf{z} = (z_1, \ldots, z_I)$, where $z_i \in \{0, 1\}$, the probability of the input is defined as [FI12, Sal15]:

$$p(\mathbf{z}) = \sum_{\mathbf{h}} p(\mathbf{z}, \mathbf{h}), \tag{4.2}$$

where the joint probability over the binary observation and hidden units is given by a Gibbs distribution:

$$p(\mathbf{z}, \mathbf{h}) = \frac{1}{\mathcal{Z}} \exp\left(-E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h})\right). \tag{4.3}$$

The energy of the distribution is given by:

$$E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h}) = -\sum_i b_i z_i - \sum_j c_j h_j - \sum_j h_j \sum_i w_{ji} z_i, \tag{4.4}$$

here $\mathbf{h} = (h_1, \ldots, h_J)^T, h_j \in \{0, 1\}$ denotes a vector of $J$ hidden units, $w_{ji}$ denotes the weighted connection between visual unit $z_i$ and $h_j$, $c_j$ is a bias term of hidden unit $h_j$ and $b_i$ is a bias term of the visual unit $z_i$. The bias terms encode the tendency for a visual or a hidden unit to be activated. We denote the full parameter set with $\boldsymbol{\lambda} = \{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$. The normalization term $\mathcal{Z}$ is obtained by marginalizing over the observations and hidden units:

$$\mathcal{Z} = \sum_{\mathbf{z}} \sum_{\mathbf{h}} \exp\left(-E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h})\right). \tag{4.5}$$

This type of model has two attractive properties: first, the introduction of the hidden units increases the expressive power of the model and allows for modeling complicated distributions, and second, because there are no connections *within*

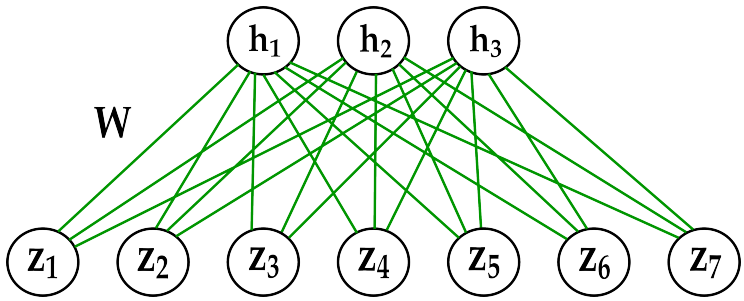



**Figure 4.3:** The basic RBM model for an input of size $(1 \times 7)$ and three hidden units. Each of the observations $\mathbf{z}$ is connected to all of the three hidden units $\mathbf{h}$ by a weighted edge.

the observation or hidden unit layers, sampling from the model is straightforward. Specifically, the conditional distributions for each hidden unit and observation are given by [FI12, Sal15]:

$$p(h_j = 1|\mathbf{z}) = \sigma(\sum_i w_{ji} z_i + c_j) \tag{4.6}$$

$$p(z_i = 1|\mathbf{h}) = \sigma(\sum_j w_{ji} h_j + b_i), \tag{4.7}$$

where $\sigma(x) = (1+exp(-x))^{-1}$. Once the model parameters $\boldsymbol{\lambda}$ have been learned, inference computations using the model are greatly facilitated by the two-layer structure where the visual units are conditionally independent of each other given the hidden units and vice versa.

**Learning the RBM parameters.** Given a training set of $M$ binary inputs $\{\mathbf{z}^m\}_{m=1}^M$, we want to adjust the model parameters such that the probability distribution fits the training data as well as possible [FI12]. This can be achieved by maximizing the log-likelihood of the model with respect to the parameters. In particular, the derivative of the log-likelihood for a single training example $m$ is given by [FI12]:

$$\frac{\partial \log p(\mathbf{z}^m)}{\partial \delta \boldsymbol{\lambda}} = -\sum_{\mathbf{h}} p(\mathbf{h}|\mathbf{z}^m) \frac{\partial E_{\mathrm{RBM}}(\mathbf{z}^m, \mathbf{h})}{\delta \boldsymbol{\lambda}} + \sum_{\mathbf{z},\mathbf{h}} p(\mathbf{z}, \mathbf{h}) \frac{\partial E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h})}{\delta \boldsymbol{\lambda}}. \tag{4.8}$$

Using the specific structure of the RBM, this can be written, for each parameter, as [FI12]:

$$\frac{\partial \log p(\mathbf{z}^m)}{\partial w_{ji}} = p(h_j = 1|\mathbf{z}^m) z_i^m - \sum_{\mathbf{z}} p(\mathbf{z}) p(h_j = 1|\mathbf{z}) z_i \tag{4.9}$$

$$\frac{\partial \log p(\mathbf{z}^m)}{\partial b_i} = z_i^m - \sum_{\mathbf{z}} p(\mathbf{z}) z_i \tag{4.10}$$

$$\frac{\partial \log p(\mathbf{z}^m)}{\partial c_j} = p(h_j = 1|\mathbf{z}^m) - \sum_{\mathbf{z}} p(\mathbf{z}) p(h_j = 1|\mathbf{z}). \tag{4.11}$$

Calculating these expressions over a batch of training data can then be niftily written as expectations [Sal15]:

$$\frac{1}{M} \sum_{m=1}^{M} \frac{\partial \log p(\mathbf{z}^m)}{\partial w_{ji}} = \mathrm{E}_{P_{\mathrm{data}}}(z_i h_j) - \mathrm{E}_{P_{\mathrm{RBM}}}(z_i h_j) \tag{4.12}$$

$$\frac{1}{M} \sum_{m=1}^{M} \frac{\partial \log p(\mathbf{z}^m)}{\partial b_i} = \mathrm{E}_{P_{\mathrm{data}}}(z_i) - \mathrm{E}_{P_{\mathrm{RBM}}}(z_i) \tag{4.13}$$

$$\frac{1}{M} \sum_{m=1}^{M} \frac{\partial \log p(\mathbf{z}^m)}{\partial c_j} = \mathrm{E}_{P_{\mathrm{data}}}(h_j) - \mathrm{E}_{P_{\mathrm{RBM}}}(h_j), \tag{4.14}$$

where $P_{\mathrm{data}}(\mathbf{z}, \mathbf{h}) = p(\mathbf{h}|\mathbf{z}) p_{\mathrm{data}}(\mathbf{z})$, and $p_{\mathrm{data}}(\mathbf{z}) = \frac{1}{M} \sum_m \delta(\mathbf{z} - \mathbf{z}^m)$ denotes the empirical data distribution [Sal15, FI12]. The latter expectation is computed with respect to the RBM distribution defined in eq. 4.3.

The main difficulty with computing the gradient is due to the expectation with respect to the model. Exact computation of the sum over *both* the visible variables $\mathbf{z}$ and hidden variables $\mathbf{h}$ is in practice intractable except for RBMs with a very small number of hidden and visible units. Due to the structure of the model, one solution would be to approximate this term by sampling. However, even if generating the samples is easy, it would take a long time to collect enough samples in order to get good approximations, which would make practical experimentation with the model quite laborious. Instead, the training is typically done using an approximation called Contrastive Divergence (CD) [Hin02]. The idea is fairly simple: instead of drawing multiple samples from the model starting from a random configuration, we only draw a single sample from the model where the sampling is initialized with the training data. Thus for a single training example the gradient in eq. 4.8 is approximated by [FI12]:

$$\mathrm{CD}(\mathbf{z}^m, \boldsymbol{\lambda}) = -\sum_{\mathbf{h}} p(\mathbf{h}|\mathbf{z}^m) \frac{\partial E_{\mathrm{RBM}}(\mathbf{z}^m, \mathbf{h})}{\partial \boldsymbol{\lambda}} + \sum_{\mathbf{h}} p(\mathbf{h}|\tilde{\mathbf{z}}) \frac{\partial E_{\mathrm{RBM}}(\tilde{\mathbf{z}}, \mathbf{h})}{\partial \boldsymbol{\lambda}},$$

where $\tilde{\mathbf{z}}$ is produced from $\mathbf{z}^m$ by sampling once from the conditional equations defined in 4.7. The intuition behind the CD approximation is that if the model

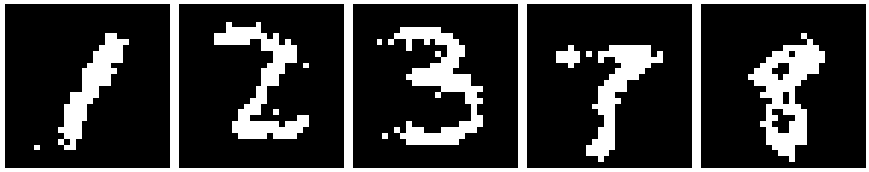


**Figure 4.4:** Random samples generated from an RBM trained on hand-written digits.

has learned the training data well, sampling once from the training data should yield a sample that closely resembles the training data which in turn gives a small gradient. However if the model does *not* represent the training data well, the sample will move away from the training example which can already give us a good idea of how to tune the parameters in order to better model the training data. The latter terms of the gradients in eq. 4.14 are thus approximated by sampling once from the training examples and then computing the expectations. Once the gradient has been approximated, the parameters are updated by taking a step in the gradient direction given a user specified step size $\eta$. In practical implementation, stochastic gradient descent is often used where the training data is divided to random partitions, or *batches*, which contain only a part of the full data set, and the gradients are approximated on each batch.

Finally to show an example, figure 4.4 displays samples generated from an RBM model which has been trained using CD on training data consisting of the famous MNIST data set[2] of hand-written digits. The model seems to have learned quite well what hand-written digits typically look like.

**Convolutional RBMs.** The basic RBM model is very good for learning distributions over small input images, for example the MNIST digits are of size $28 \times 28$. However, extending the basic RBM to model large images is problematic as noted in [LGRN09, NRM09]. This is mainly due to two reasons: first, the basic model does not scale well to full-sized images due to the number of parameters to be learned[3], and second, it ignores the translation-invariance of large images where many features can be present in different parts of the image [LGRN09, NRM09]. In our case the inputs are manually annotated lesion maps cropped to roughly of size $120 \times 120 \times 120$, so the basic RBM model is not suitable for modeling images this large. To overcome this, we adopt a convolutional version of the RBM introduced in [LGRN09, NRM09]. The energy

[2] Available at: `http://yann.lecun.com/exdb/mnist/`

[3] For a 3D volume of size $N \times N \times N$ we would have $N^3$ weights per hidden unit

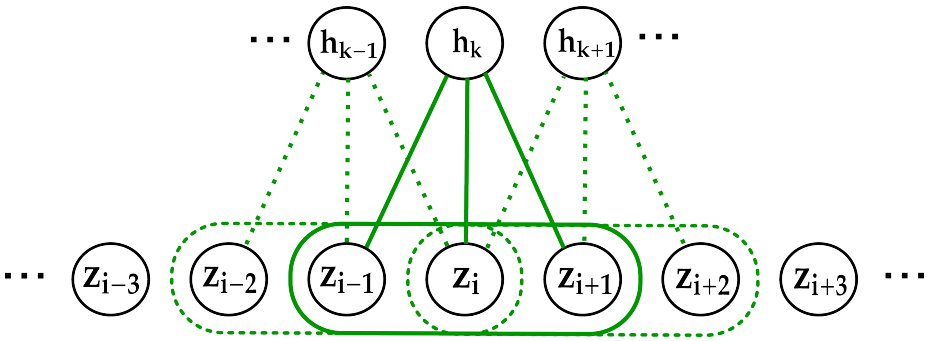



**Figure 4.5:** A single convolutional RBM unit. A weight vector is convolved over the input $\mathbf{z}$ and at each step the weights are connected to a single hidden unit in the group. This unit encodes if a feature was detected in this part of the input.

of the RBM model is now defined as [LGRN09]:

$$E_{\text{cRBM}}(\mathbf{z}, \mathbf{H}) = -\sum_i b_i z_i - \sum_k c_k \sum_j h_j^k - \sum_k \mathbf{h}^k \bullet (\mathbf{w}_k * \mathbf{z}), \tag{4.15}$$

where $*$ denotes a convolution and $\bullet$ denotes an element-wise product followed by a summation. The hidden layer now consists of $K$ groups $\mathbf{H} = \{\mathbf{h}^1, \ldots, \mathbf{h}^K\}$, where each group is a binary vector $\mathbf{h}^k$ of size $N_h$. Each group is associated with a weight vector, or filter, $\mathbf{w}_k$ which is of size $N_w$. Thus, if the input is of size $N_I$, the size of each hidden group is defined to be $N_h = N_I - N_w + 1$. In this model, each hidden group shares a bias term $c_k$ which encodes the tendency for a feature to be present in the image. A graphical representation of the model for a single convolutional RBM unit, i.e., where $K = 1$, is shown in fig. 4.5. The conditional probabilities factorize nicely as before, and for each hidden unit and observation we have [LGRN09]:

$$p(h_j^k = 1|\mathbf{z}) = \sigma((\tilde{\mathbf{w}}_k * \mathbf{z})_j + c_k) \tag{4.16}$$

$$p(z_i = 1|\mathbf{H}) = \sigma((\sum_k \mathbf{w}_k * \mathbf{h}^k)_i + b_i), \tag{4.17}$$

where $\tilde{\mathbf{w}}$ denotes that the weights are flipped [LGRN09].

As a concrete toy example consider a binary input of size $(1 \times 5)$ and one filter $\mathbf{w} = [2, 2, -5]$ of size $(1 \times 3)$, see fig. 4.6 for a visualization. Here the filter now encodes a feature, which looks for clustered groups of two pixels. The filter is convolved over the input producing a response of size $(1 \times 3)$. The group of hidden units $\mathbf{h}$ for this filter is a vector of the same size as the response, where the probability for a hidden unit at position $j$ to be turned on

is $p(h_j = 1|\mathbf{z})$. Thus each hidden unit in the group tells if a feature encoded by the filter was detected in a given part of the image. Note, that the model presented in [LGRN09] assumes a single bias over the whole image whereas we use a bias term per voxel. This allows us to include the spatial distribution of lesions into the model.

Due to the similar structure of the convolutional RBM to the basic RBM, the model parameters can be learned in exactly the same way for both models. However, instead of the basic CD approach, in this work we have used the so-called persistent contrastive divergence algorithm (PCD) [Tie08]. In PCD, the sampling chain is not re-initialized with a training example when the gradient is computed as in CD, but a persistent sampling chain is kept running throughout the whole parameter learning process. At the start of the learning, the chains are initialized to zero, and a sample from each chain is drawn at each gradient step without re-setting it. The idea is that if the parameter updates are small the model does not change that much, and the samples should stay close to the model distribution [Tie08]. PCD has been shown to learn better models compared to the basic CD framework [Tie08].

Finally to conclude the section, we show some samples from a convolutional RBM model trained using manually annotated lesion maps. Figure 4.7 shows examples of training data along with samples drawn from the learned model. Another interesting example is shown in fig. 4.8, where we sampled from the model with an input lesion map where part of the image was set to zero. It can be seen from the figure, that the model can quite well "imagine" what the missing part might have been.

### 4.2.2 Joint segmentation prior for brain anatomy and lesions

To incorporate the convolutional RBM lesion model into the healthy brain segmentation framework, we need to define a new segmentation prior that takes into account both terms. Recall that in chapter 3, the segmentation prior was written as:

$$p(\mathbf{l}) = \int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{x}.$$

A simple way to include the lesion model into the framework would be to assume that lesions can appear anywhere in the brain. In essence, this would mean removing the edge between the healthy labels $\mathbf{l}$ and lesions $\mathbf{z}$ in the graphical model shown in fig. 4.2. The healthy neuroanatomical labels and lesions would thus be generated separately from their respective distributions, and the segmentation prior would simply be: $p(\mathbf{l}, \mathbf{z}) = p(\mathbf{l})p(\mathbf{z})$. However, we want to

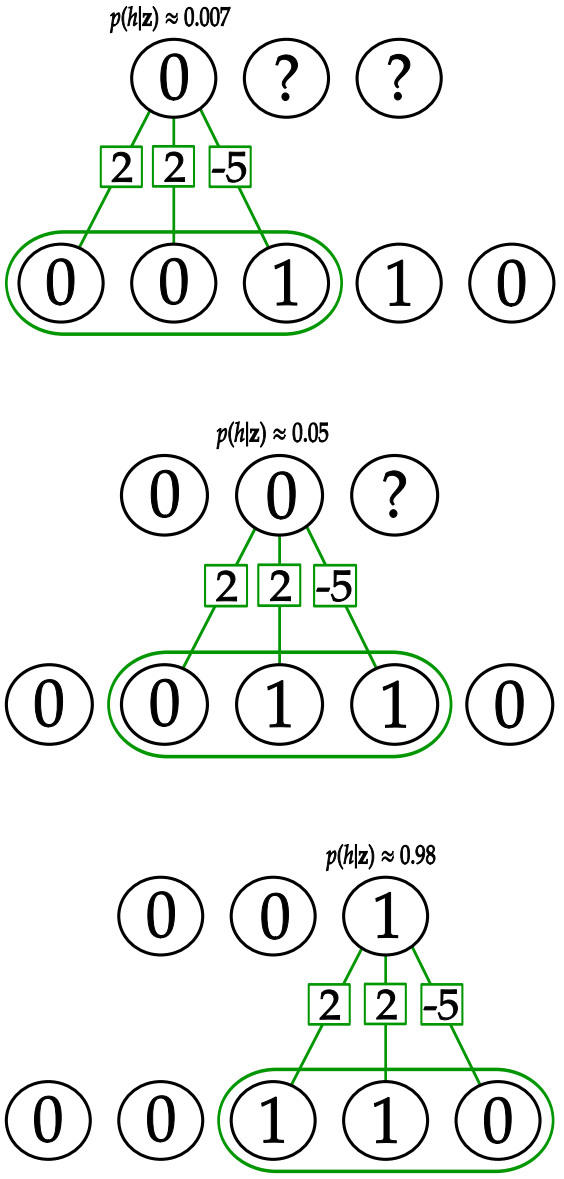



**Figure 4.6:** A toy example of a convolutional RBM. A weight vector is convolved over the input at each step producing a response. The hidden units are activated depending on the size of the response.

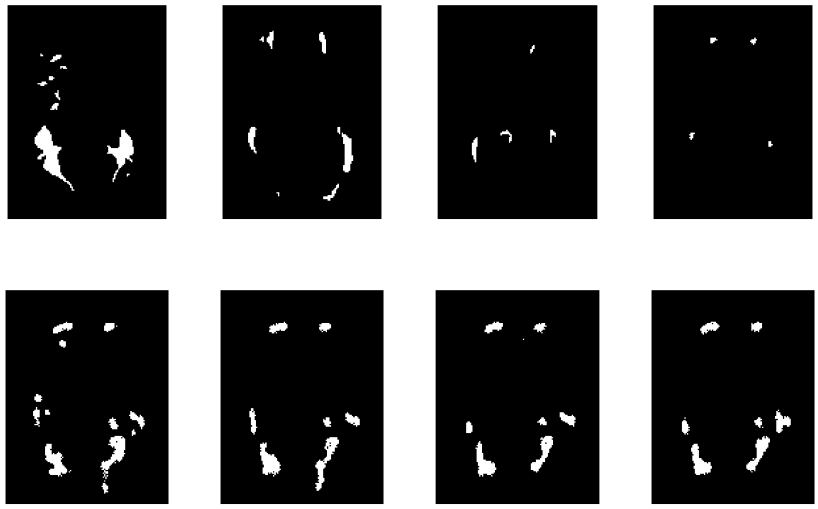


**Figure 4.7:** Samples from a convolutional RBM trained on binary lesion maps. Top row: training examples. Bottom row: random samples from the model.

enforce that lesions can only appear *within* white matter structures. To achieve this, we define a joint Gibbs distribution over $\mathbf{l}$, $\mathbf{z}$ and $\mathbf{H}$, which is conditioned on the mesh node positions $\mathbf{x}$:

$$p(\mathbf{l}, \mathbf{z}, \mathbf{H}|\mathbf{x}) \propto \exp\left(-E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{H}) + \sum_i \log p_i(l_i|\mathbf{x}) - \sum_i \alpha(l_i, z_i)\right), \quad (4.18)$$

where in abuse of notation $p_i(l_i|\mathbf{x})$ refers to the deformable segmentation prior in section 3.1.1.3, and $\alpha(l_i, z_i)$ is defined as:

$$\alpha(l, z) = \begin{cases} 0, & \text{if} \quad l = \mathrm{wm} \quad \text{or} \quad z = 0 \\ \infty, & \text{otherwise} \end{cases}$$

which now enforces the restriction that the lesions can only occur when the underlying neuroanatomical structure belongs to the white matter tissue class[4]. The joint segmentation prior is then obtained as:

$$p(\mathbf{l}, \mathbf{z}) = \int_{\mathbf{x}} p(\mathbf{l}, \mathbf{z}|\mathbf{x}) p(\mathbf{x}) \mathrm{d}\mathbf{x}, \quad (4.19)$$

[4]Referring back to table 3.1 in chapter 3, this would include WM, CWM and BS.

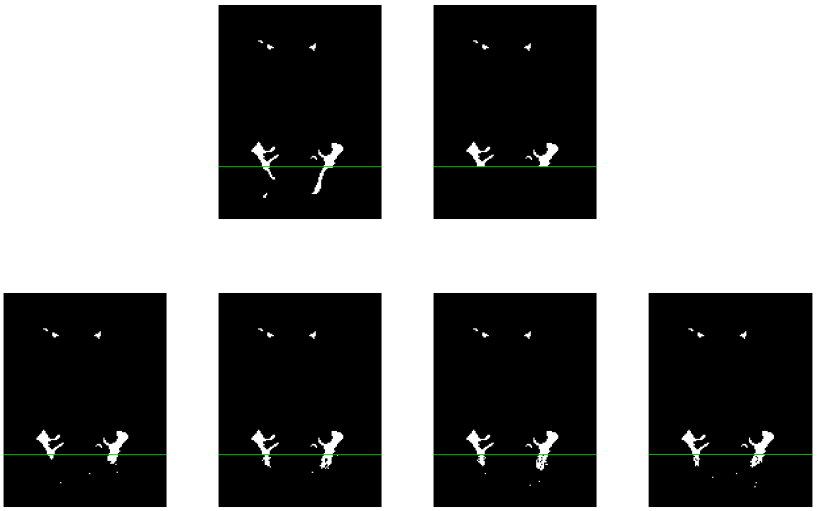


**Figure 4.8:** Example of how the learned model can "imagine" lesion shapes. Top row: full lesion map and a lesion map where the lower part is set to zero. Bottom row: samples from the model trying to fill in the missing part.

where:

$$p(\mathbf{l}, \mathbf{z}|\mathbf{x}) = \sum_{\mathbf{H}} p(\mathbf{l}, \mathbf{z}, \mathbf{H}|\mathbf{x}). \tag{4.20}$$

### 4.2.3 Likelihood function

Similarly we need to extend the likelihood model to be conditioned on both the neuroanatomical structures $\mathbf{l}$ and lesions $\mathbf{z}$. In this work, similar to some of the unsupervised lesion segmentation approaches described in the introduction, we model the lesions explicitly with their own Gaussian mixture model. However, as mentioned before, lesions do not have a well-defined intensity profile. For example some lesions appear hyperintense in T2-weighted and FLAIR images when compared to normal white matter, but can appear as either iso- or hypointense to normal white matter in T1-weighted images. To account for this ambiguity in the intensity profile we model lesions with Gaussian distributions with large variances.

Recall that the full likelihood function in chapter 3 was written as:

$$p(\mathbf{D}|\mathbf{l}) = \int_{\boldsymbol{\theta}} p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}.$$

Now, we extend this to:

$$p(\mathbf{D}|\mathbf{l}, \mathbf{z}) = \int_{\boldsymbol{\theta}} p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}, \tag{4.21}$$

where

$$p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \boldsymbol{\theta}) = \prod_i p(\mathbf{d}_i|l_i, z_i, \boldsymbol{\theta}). \tag{4.22}$$

Here we again assume that the intensities in the voxels are independent of each other given the labels. The GMM is now defined as:

$$p_i(\mathbf{d}|l, z, \boldsymbol{\theta}) = \sum_{g=1}^{G_l} w_{lg} \mathcal{N}(\mathbf{d} - \mathbf{b}|\boldsymbol{\mu}_{lg}, \gamma^z \boldsymbol{\Sigma}_{lg}). \tag{4.23}$$

Here we have introduced a user-specified factor $\gamma >> 1$ as a multiplier to the covariance matrix. Note here that when no lesion is present ($z = 0$), the model reduces to the same likelihood model as was used in chapter 3, but when a lesion is observed ($z = 1$) it is modeled with a GMM which shares its parameters with the healthy (white matter) structures, but has a larger variance.

## 4.3 Inference

We are again faced with the problem that the segmentation posterior:

$$p(\mathbf{l}, \mathbf{z}|\mathbf{D}) \propto p(\mathbf{D}|\mathbf{l}, \mathbf{z}) p(\mathbf{l}, \mathbf{z}),$$

can not be easily evaluated due to the marginalizations over the parameters. Thus, as in chapter 3, we resort to the empirical Bayes approximation. The posterior of the parameters $\{\boldsymbol{\theta}, \mathbf{x}\}$ is written as:

$$\begin{aligned} p(\boldsymbol{\theta}, \mathbf{x}|\mathbf{D}) &\propto p(\mathbf{D}|\boldsymbol{\theta}, \mathbf{x}) p(\mathbf{x}) p(\boldsymbol{\theta}) \\ &= \left( \sum_{\mathbf{l},\mathbf{z}} p(\mathbf{D}, \mathbf{l}, \mathbf{z}|\boldsymbol{\theta}, \mathbf{x}) \right) p(\mathbf{x}) p(\boldsymbol{\theta}) \\ &= \left( \sum_{\mathbf{l},\mathbf{z}} p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \boldsymbol{\theta}) p(\mathbf{l}, \mathbf{z}|\mathbf{x}) \right) p(\mathbf{x}) p(\boldsymbol{\theta}). \end{aligned}$$

However, now the expression does not factorize over the voxels as was the case before, due to the cRBM model which introduces dependencies between the voxels. To overcome this difficulty, we temporarily replace the cRBM lesion prior with a constant spatial prior for the time of the parameter estimation. Although this might seem odd after we spent a lot of effort devising the lesion model, at this point we are not looking for accurate lesion segmentations but are trying to obtain estimates for the parameters in a robust manner. The temporary lesion prior has the following form:

$$p(\mathbf{l}, \mathbf{z}|\mathbf{x}) \propto \exp\left(-E_{\mathrm{TMP}}(\mathbf{z}, \mathbf{l}) + \sum_i \log p(l_i|\mathbf{x}) - \sum_i \alpha(l_i, z_i)\right), \quad \text{with}$$
$$E_{\mathrm{TMP}}(\mathbf{z}, \mathbf{l}) = -\sum_i [l_i = \mathrm{wm}]\,(z_i \log(\pi) + (1 - z_i)\log(1 - \pi)),$$

where $0 \leq \pi \leq 1$ is a user-defined constant spatial prior for the lesions *within* white matter. For non-white matter structures the model now reduces to the likelihood model from chapter 3, whereas for white matter structures the likelihood is now given by:

$$p_i(\mathbf{d}|l = \mathrm{wm}, \boldsymbol{\theta}) = \sum_{g=1}^{G_l} w_{lg} \left((1 - \pi)\mathcal{N}(\mathbf{d} - \mathbf{b}|\boldsymbol{\mu}_{lg}, \boldsymbol{\Sigma}_{lg}) + \pi\mathcal{N}(\mathbf{d} - \mathbf{b}|\boldsymbol{\mu}_{lg}, \gamma\boldsymbol{\Sigma}_{lg})\right),$$

yielding a distribution with heavier tails, as shown in fig. 4.9, which makes the parameter estimation more robust in the presence of lesions. Now the parameter estimation proceeds similarly as in chapter 3 by alternating optimization of the mesh node positions and intensity model parameters. The only modification is that when updating the weights, means and covariances of the white matter structures, we need to take into account that each of the mixture components now consists of two Gaussians with shared parameters. The modified update equations of the weights, means and covariances for a combination of the two white matter Gaussians ($k = \mathrm{wm}$) can be written as:

$$w_{k,g} \leftarrow \frac{\sum_{i=1}^{I}(q_i^{k,g,1} + q_i^{k,g,2})}{\sum_{i=1}^{I}\sum_{g'=1}^{G_k}(q_i^{k,g',1} + q_i^{k,g',2})},$$
$$\boldsymbol{\mu}_{k,g} \leftarrow \frac{\sum_{i=1}^{I}(q_i^{k,g,1} + \gamma^{-1}q_i^{k,g,2})(\mathbf{d}_i - \mathbf{b}_i)}{\sum_{i=1}^{I}(q_i^{k,g,1} + \gamma^{-1}q_i^{k,g,2})},$$
$$\boldsymbol{\Sigma}_{k,g} \leftarrow \frac{\sum_{i=1}^{I}(q_i^{k,g,1} + \gamma^{-1}q_i^{k,g,2})(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{b}_i)(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{b}_i)^T}{\sum_{i=1}^{I}(q_i^{k,g,1} + q_i^{k,g,2})},$$

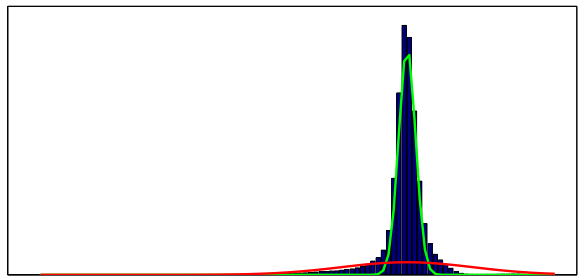


**Figure 4.9:** Example of the robust intensity model: the intensity histogram of the white matter tissue class where the Gaussian modeling healthy white matter is overlaid in green and the Gaussian modeling lesions in red.

where the soft assignment are defined as:

$$q_i^{k,g,1} \leftarrow \frac{(1-\pi) w_{k,g} \mathcal{N}\left(\mathbf{d}_i - \mathbf{b}_i | \boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}\right) p_i(k|\mathbf{x})}{\sum_{k'=1}^{K} p_i(\mathbf{d}_i|k', \boldsymbol{\theta}) p_i(k'|\mathbf{x})}$$

$$q_i^{k,g,2} \leftarrow \frac{\pi w_{k,g} \mathcal{N}\left(\mathbf{d}_i - \mathbf{b}_i | \boldsymbol{\mu}_{k,g}, \gamma\boldsymbol{\Sigma}_{k,g}\right) p_i(k|\mathbf{x})}{\sum_{k'=1}^{K} p_i(\mathbf{d}_i|k', \boldsymbol{\theta}) p_i(k'|\mathbf{x})}.$$

Here the fixed covariance factor $\gamma$ now shows up as a multiplier in the updates for the means and covariances. Note that because both the means and (co)variances of the two Gaussians are tied, only the factor $\gamma$ appears in the update equation for the means. If there was no restriction on the variances but the means were restricted to be equal, i.e., we would have a scale mixture, the mean update would depend on the variances as well [CH13].

Once we have obtained the parameter estimates $\{\hat{\boldsymbol{\theta}}, \hat{\mathbf{x}}\}$, we replace the simple temporary prior with the proper cRBM model. The approximated segmentation posterior is now written as:

$$p(\mathbf{l}, \mathbf{z}|\mathbf{D}) \approx p(\mathbf{l}, \mathbf{z}|\mathbf{D}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{x}}),$$

where the intractable integrals are gone. This posterior still does not factorize over the voxels, and thus assigning the most probable class (including lesions) in each voxel independently is not possible. However, we can exploit the specific two-layer structure of the cRBM model to generate samples from the posterior.

The sampling can be done using block-Gibbs sampling in two steps: first, we sample the hidden units $\mathbf{H}$ given the lesions $\mathbf{z}$, and then we sample jointly from

$\mathbf{l}$ and $\mathbf{z}$ given the sampled values for the hidden units, the target data and the optimal parameter values. The first step takes the form (see eq. 4.17)

$$p(h_j^k = 1|\mathbf{z}) = \sigma((\tilde{\mathbf{w}}_k * \mathbf{z})_j + c_k), \tag{4.24}$$

where the values of the hidden units $\mathbf{H}$ can be sampled independent of each other given the lesions. For the second step we obtain:

$$p(l_i, z_i|\mathbf{d}_i, \hat{\boldsymbol{\theta}}, \hat{\mathbf{x}}, \mathbf{H}) \propto \begin{cases} p(\mathbf{d}_i|l_i, z_i, \hat{\boldsymbol{\theta}})p(l_i|\hat{\mathbf{x}})p(z_i|\mathbf{H}), & \text{if } l_i = \text{wm or } z_i = 0 \\ 0, \quad \text{otherwise.} \end{cases} \tag{4.25}$$

where the lower equality stems from the prior which prevents lesions from appearing outside of white matter structures.

We collect $S$ samples $\{\mathbf{l}^s, \mathbf{z}^s\}_{s=1}^S$, by alternate sampling from the distributions defined in eqs. 4.24 and 4.25. Note that the values of the hidden units $\mathbf{H}$ are of no interest to us and are discarded. The sampling is initialized by estimating the lesion segmentation using the temporary uniform spatial prior and the estimated parameter values $\{\hat{\boldsymbol{\theta}}, \hat{\mathbf{x}}\}$ from the approximate segmentation posterior. Given $S$ samples, we approximate the maximum-a-posteriori segmentation by using voxel-wise majority voting across $\{\mathbf{l}^s\}$ and $\{\mathbf{z}^s\}$ which gives us the final "hard" segmentation.

## 4.4 Experiments and results

In this section we will briefly present the experiments and results from paper C. In order to demonstrate our lesion segmentation approach, we tested it on the 20 publicly available training cases of the MICCAI 2008 challenge on multiple sclerosis lesion segmentation [SLC+08]. We also compared the proposed method against two other state-of-the-art lesion detection methods [GCM+11, WRR13], both of which have been shown to give greatly improved segmentation results compared to the challenge winner. The first one is a supervised approach based on a random forest classifier [GCM+11], and the second one is an unsupervised approach based on patch-based dictionary learning. Next, we will give a brief overview of both methods.

### 4.4.1 Benchmark methods

**Random forest classifier.** The RF classifier combines many independent decision tree classifiers to produce a final segmentation [GCM+11]. The power of

this approach comes from using an ensemble of simple classifiers, which individually might not yield accurate classifications, but can perform very well when combined. In particular, each classification tree is trained with a random subset of the training data. Furthermore, at each node of a tree only a random subset of all the extracted features is available for optimization [GCM+11]. This random sampling improves the generalization properties of the RF to unseen data [GCM+11]. The full set of features consists of a collection of local features, such as the intensity of a voxel, and context-rich features, such as comparing the intensity of a voxel to the mean intensity of a patch sampled randomly in a large neighborhood of the given voxel. Once the RF has been trained, the leaf nodes of each tree contain a probability of the voxel belonging to lesion or background. This probability is computed during training as the fraction of training samples labeled as lesion, or background, that end up in the given leaf node. Lesions in a target scan are then detected by propagating each voxel through each tree, collecting the probabilities at the leaf nodes and finally classifying each voxel based on the averaged probability over all the trees.

**Patch-based dictionary learning.** The method presented in [WRR13] is based on sparse patch-based dictionary learning (DL). This approach builds upon the unsupervised models where lesions are found as outliers compared to normal brain tissue. Given an input image, first the area of interest, typically consisting of all brain tissues where non-brain structures have been masked out, is divided into non-overlapping patches. Next a sparse dictionary is learned from these patches by searching for an optimal subset of patches which minimize the reconstruction error between each patch and the subset. This learned subset of patches then becomes the patch dictionary. Finally, lesions are found by computing the reconstruction error between each image patch and the optimal dictionary. The idea is that patches with a large reconstruction error represent possible lesions. The final reconstruction error map is then thresholded to obtain a lesion segmentation. Note that this method is unsupervised as the patch library is learned from the target scan. However, a small amount of labeled training data is needed to choose the optimal reconstruction error threshold level.

### 4.4.2 Data

The dataset includes 10 subjects scanned at the Children's Hospital Boston (CHB) and another 10 subjects scanned at the University of North Carolina (UNC), and consists of T1-weighted, T2-weighted and FLAIR scans with isotropic resolution of 0.5mm, along with expert segmentations provided by CHB. Expert segmentations from UNC are currently also available, but during the time of the

challenge this was not the case, thus we also only use the CHB segmentations for training to ensure a fair comparison [SLC+08]. The volumes are of size $512 \times 512 \times 512$, so we downsampled them by a factor of two as is often done for this data set [WRR13, GCM+11].

### 4.4.3 Implementation

Because we only had 20 manual segmentations available for training the RBM model, which is quite a small number, we decided to augment the dataset by applying two rotations of 10 and $-10$ degrees around the three main axes. This resulted in 6 extra training scans per subject for a total of 140 manual segmentations in the augmented dataset. We trained different RBM models with either $K = 20$ or $K = 40$ hidden unit groups and with filter sizes of $(N_w \times N_w \times N_w)$, where $N_w$ was either 5, 7 or 9. Each model was trained with 5600 gradient steps in the PCD algorithm [Tie08]. The training time for each model was approximately 3 days using a Matlab implementation on a machine with a GeForce GTX Titan 6GB GPU.

Based on pilot experiments, we found that using two mixture components for white matter worked well (i.e., $G_{\text{wm}} = 2$), provided that one of the Gaussians is constrained to be a near-uniform distribution that can collect model outliers other than white matter lesions (in practice we use a Gaussian with a fixed scalar covariance matrix $10^6\mathbf{I}$ and weight 0.05). Finally, as the main characteristic of white matter lesions is that they appear hyper-intense compared to normal white matter in FLAIR contrast [GLFN+13], we decided to only allow voxels to be assigned to lesion in the Gibbs sampling process if their intensity is higher than the estimated white matter mean in FLAIR.

The segmentation algorithm was implemented in Matlab, except for the mesh deformation part, which was written in C++, and the RBM convolutions, which were performed on a GPU. In our experiments, estimation of the parameters $\{\hat{\boldsymbol{\theta}}_d, \hat{\mathbf{x}}\}$ was performed on a cluster where each node has two quad-core Xeon 5472 3.0GHz CPUs and 32GB of RAM. Only one core was used in the experiments, taking roughly 1.7 hours per subject. Gibbs sampling was again done on a machine with a GeForce GTX Titan 6GB GPU. We generated $S = 150$ samples, collected after an initial burn-in of 50 sampling steps, taking approximately 10 minutes per subject. Thus the full segmentation time for a single target scan is roughly two hours.

### 4.4.4 Evaluation set-up

As the evaluation metrics, we used the voxel-wise true positive rate TPR = $\frac{\text{TP}}{\text{TP+FN}}$ and the positive predictive value PPV = $\frac{\text{TP}}{\text{TP+FP}}$. Here TP, FP and FN count the true positive, false positive and false negative voxels compared to the expert segmentation. This allows us to compare our results to the ones reported in [GCM+11] and [WRR13]. Because we need data both to train the cRBM model and to tune the parameters of the model, i.e., the number of hidden groups $K$, the size of the filters $N_w$, the width factor of the lesion Gaussians $\gamma$ and the temporary spatial prior $\pi$, we performed the evaluation in a cross-validation setting. In particular, we divided the available data set of 20 subjects randomly into five distinct sets, each having 16 training subjects and 4 test subjects. In each of the five groups, only the training subjects were used to train the lesion shape models and to tune the model parameters, whereas the 4 subjects were only used for testing. Using each 16 subject set, we trained a cRBM model using the different filter sizes and number of hidden unit groups specified in the implementation section. For the likelihood parameters we used the following values: $\gamma = \{5, 10, 20, 40, 100\}$ and $\pi = \{0.1, 0.2, 0.3, 0.4, 0.5\}$. The optimal parameter combination $(\gamma, \pi, N_w, K)$ was then found on the 16 subjects by searching for the combination which maximizes the product of the mean TPR and PPV over the subjects. This parameter combination was then used for segmenting the 4 test subjects. We decided to use the product of the TPR and PPV as a measure of fitness as it encourages parameter combinations that do not over- or under-segment the lesions.

### 4.4.5 Results

The TPR and PPV scores for the different lesion segmentation methods are shown in table 4.1. On average the proposed method clearly outperforms the also unsupervised patch-based dictionary learning approach, and achieves slightly better average scores than the supervised random forest classifier. Note that all of the three methods perform better than the MICCAI 2008 challenge winner, which was based on an unsupervised GMM approach [SLAM08], and achieved an average TPR of 0.21 and an average PPV of 0.30. In figure 4.10 example segmentations from three subjects in the data set are shown.

| | DL [WRR13] | | RF [GCM+11] | | Proposed | |
|---|---|---|---|---|---|---|
| Patient | TPR | PPV | TPR | PPV | TPR | PPV |
| CHB01 | 0.60 | 0.58 | 0.49 | **0.64** | **0.75** | 0.57 |
| CHB02 | 0.27 | 0.45 | 0.44 | **0.63** | **0.57** | 0.48 |
| CHB03 | 0.24 | 0.56 | 0.22 | 0.57 | **0.30** | **0.69** |
| CHB04 | 0.27 | 0.66 | 0.31 | **0.78** | **0.59** | 0.49 |
| CHB05 | 0.29 | 0.33 | 0.40 | **0.52** | **0.45** | 0.39 |
| CHB06 | 0.10 | 0.36 | **0.32** | **0.52** | 0.19 | 0.50 |
| CHB07 | 0.14 | 0.48 | **0.40** | 0.54 | 0.34 | **0.65** |
| CHB08 | 0.21 | **0.73** | **0.46** | 0.65 | 0.37 | 0.70 |
| CHB09 | 0.05 | 0.22 | **0.23** | 0.28 | 0.04 | **0.55** |
| CHB10 | 0.15 | 0.12 | **0.23** | 0.39 | 0.19 | **0.69** |
| Mean | DL [WRR13] TPR=0.33 | DL [WRR13] PPV=0.37 | RF [GCM+11] TPR=0.40 | RF [GCM+11] PPV=**0.40** | Proposed TPR=**0.41** | Proposed PPV=**0.40** |

| | DL [WRR13] | | RF [GCM+11] | | Proposed | |
|---|---|---|---|---|---|---|
| Patient | TPR | PPV | TPR | PPV | TPR | PPV |
| UNC01 | **0.33** | **0.29** | 0.02 | 0.01 | 0.02 | 0.01 |
| UNC02 | 0.54 | **0.51** | 0.48 | 0.36 | **0.75** | 0.29 |
| UNC03 | **0.64** | **0.27** | 0.24 | 0.35 | 0.28 | 0.19 |
| UNC04 | 0.40 | **0.51** | 0.54 | 0.38 | **0.62** | 0.40 |
| UNC05 | 0.25 | 0.10 | **0.56** | **0.19** | 0.50 | 0.18 |
| UNC06 | 0.13 | **0.55** | 0.15 | 0.08 | **0.17** | 0.10 |
| UNC07 | 0.44 | 0.23 | **0.76** | 0.16 | 0.60 | **0.26** |
| UNC08 | 0.43 | 0.13 | **0.52** | **0.32** | 0.27 | 0.21 |
| UNC09 | **0.69** | 0.06 | 0.67 | **0.36** | 0.67 | 0.21 |
| UNC10 | 0.43 | 0.23 | **0.53** | 0.34 | 0.47 | **0.48** |

**Table 4.1:** Quantitative comparison with two state-of-the-art methods.

**Figure 4.10:** Example of scans and segmentations for the MICCAI data set. Each row shows one subject. From left to right: T1-weighted scan, T2-weighted scan, FLAIR scan, manual segmentation overlaid on the FLAIR scan and the joint lesion and structural segmentation from our approach. Lesions are highlighted in red.

## 4.5 Discussion

The results show that our generative lesion approach compares well to current state-of-the-art methods in lesion segmentation performance, while being sequence-adaptive and also segmenting the surrounding neuroanatomy to 39 different cortical and sub-cortical structures. The MICCAI 2008 challenge dataset only includes manual segmentations of lesions, so validation of the automatic segmentations of the healthy structures could not be performed. However, visual inspection of the 20 cases did not show any significant failures in the whole-brain segmentation component of the method. We note that neither of the benchmark methods segments other structures than the lesions. Furthermore, the RF classifier, being a supervised approach, is specifically trained on the contrast-properties of this particular dataset, and is thus less generally applicable than the proposed method and the patch-based dictionary learning approach.

Also worth mentioning is that the scores reported by the unsupervised dictionary learning approach [WRR13] are not entirely comparable to the other two approaches on the UNC subjects, because in [WRR13] the authors have used the segmentations from the expert rater from UNC for validating these subjects. This explains the significantly better scores reported on the UNC01 subject by the dictionary learning approach. As mentioned before, at the time of the challenge only segmentations from CHB were available, so we have only used these for training and validation. However, this brings us to one of the main issues in validating the automatic methods against ground truth data: even manual segmentations of the same subject have a large variability. The two ground truths from CHB and UNC show an overlap of only 0.68 for the MICCAI data set [GLFN$^+$13]. This variation is due to the somewhat ambiguous definition of the white matter lesions as "Areas that are hyperintense with respect to normal-appearing white matter on T2w or FLAIR images that are not due to normal structures." [GLFN$^+$13]. As no specific intensity-limit has been defined, it is up to the expert raters to interpret where such a limit should be placed. However, for validation purposes comparing against the ground truth is somewhat unavoidable, but it should be kept in mind that the gold standard might not be perfect in itself.

Even though the proposed framework already performs well, there is still room for improvement. Next we will discuss some difficulties encountered during the project related to learning and applying the modeling framework, and discuss how these problems could be solved.

**Difficulties related to the cRBM model.** One of the main difficulties in the modeling framework is related to learning a good cRBM model of the

lesions. Because the cRBM model has a lot of tunable parameters, it is very prone to over-fitting especially with the small amount of training data we have. Although in this work we augmented the dataset by rotating the training data to produce 112 labelings to train on for each 16 subject training set, a cRBM model with filter size $7 \times 7 \times 7$, 20 hidden groups and a visual bias term of size $120 \times 120 \times 120$ still has in total $7^3 * 20 + 20 + 120^3 = 1734880$ parameters to tune. Interestingly the shape model configuration that was chosen in all cross-validation rounds was the one with filter size $N_w = 5$ and $K = 20$ hidden groups, which is the model with the fewest parameters. As all the different shape models were trained for the same number of gradient steps, this implies that the models with more parameters have either over-fitted to the training data or that the learning did not converge to a good solution. The bias term of the visual units can be especially problematic, as some of the voxels might not have a lesion in any of the training samples. The model could thus learn that this particular voxel should never have a lesion. This is similar to the over-fitting problem we discussed with relation to the standard probabilistic atlases in chapter 3. A related problem is that we can not track the development of the log-likelihood of the model during learning, because the normalization term can not be computed in practice. Therefore, we can not know when the learning has converged, or compare the different model configurations. The shape models thus have to be validated by either looking at the samples they produce or measuring segmentation performance of the different configurations after the model parameters have been learned.

These problems can, however, be overcome or at least alleviated. A straightforward solution to the over-fitting problem would be to acquire more expert labelings, but as discussed before this is very time consuming and costly. Therefore it is unlikely to ever have as large training sets as are used to train some of the deep learning models[5], in a medical image analysis task. However, it should be possible to obtain a training set on the order of a hundred expert lesion segmentations. This would already be very helpful for training the shape model. A more immediate solution would be to restrict the number of parameters related to the visual bias, which includes most of the model parameters. In [LGRN09], where the cRBM model is introduced, the authors use a shared visible bias over the full observation layer. This approach might not be optimal for our task as the voxel-wise bias term helps to encode probable locations for the lesions, but either enforcing parameter sharing between neighboring voxels or including a smoothness prior on the visual bias parameters would likely be helpful. In the case of non-convolutional RBMs a L2-penalty is often added to the gradient of the filters in order to limit the filter values for becoming too large [Hin12]. The same approach could be used on the visual bias parameters, so that values of

[5]The famous ImageNet database aimed at training models for image classification has over 14 million examples.

the bias would not become too negative (never a lesion) or too positive (always a lesion). All these approaches result in a lower effective number of parameters to learn, while still encoding enough information about the spatial distribution of the lesions. In [EHWW13] the authors show that stacking more hidden layers on top of each other in the RBM model can also help in avoiding over-fitting. However, the model proposed in the article does not readily scale to large images, but it would definitely be worthwhile to experiment with a model with more than just one hidden layer.

Regarding the training of RBM models, a lot of research has been put into how the training can be done in a robust manner. This has mainly focused on improving the approximation of the model expectation term in the gradient expression. There are two problems related to the CD gradient approximation: first, because the sampling is initialized with a training example the samples do not come from the "true" model distribution, and second, if the absolute values of the parameters become very large, the sampling chain will mix very slowly and learning is stagnated [FI12]. The PCD learning algorithm, used in this work, tries to address the bias problem by not initializing the sampling chain with the learning samples, but might still suffer from low mixing rates of the chain. Good results have been obtained using a so-called parallel tempering (PT) learning algorithm [DCB+10], where multiple smoothed replicas of the true model sampling chain are run in parallel[6]. Swaps between the chains are done randomly which increases the mixing of the chains. However, this approach is computationally, and especially memory-wise, quite demanding due to the many sampling chains that need to be updated. One interesting thing to try, would be to estimate the log-likelihood of the model while learning. Although computing the normalization term is intractable, it can be approximated using annealed importance sampling (AIS) [Sal08]. Even if the approximation would not be perfect, it could give us some kind of idea whether the learning has converged or not.

**Difficulties related to the likelihood model.** Exact modeling of the lesion intensities is very difficult due to their varied appearance within the MR scans. This can be seen in fig. 4.10 middle row, where some lesion in the FLAIR scan are very bright whereas other are darker. Lesions close to gray matter structures can easily be assigned to gray matter because both appear brighter than white matter in the FLAIR scans. Furthermore, the FLAIR scans exhibit hyperintense regions close to the ventricles [NGS+09], which have appearance similar to lesions and can lead to false positive detections. The proposed framework can reduce the false positive detections related to gray matter structures due to the detailed probabilistic atlas, but the hyperintense artifacts can sometimes be

[6] In similar fashion to simulated annealing.

very hard to differentiate from lesions.

In [TW15], the authors propose to alleviate the problem related to the overlapping intensity profiles of lesions and healthy tissues by combining subject-specific intensity models with prior information in the form of a local intensity-atlas learned from training scans. Although this is shown to be beneficial in the article, we would like our model to be fully unsupervised so that it can be readily used on MR images acquired with different scanners or scan sequences, and thus this is not a feasible approach. However, some prior information related to the lesion intensities can be included into the model. As discussed in the implementation section, the lesions typically appear hyper-intense compared to normal white matter in FLAIR-sequences. In this work, we exploited this information when sampling from the model by allowing voxels to be labeled as lesion only if their intensity was higher than the estimated mean white matter intensity in FLAIR. This restriction could also be built into the model, using, for example, truncated or skew-normal distributions [PHW$^+$09]. The skew-normal distributions have an extra parameter controlling the "skewness" of the distribution, which pushes the distribution towards either side of the mean value dependent on the parameter. We have done some initial tests with replacing the Gaussian lesion likelihood function with a multivariate skew-normal distribution, and it seems it might be a better model of lesion intensities although further experiments are needed.

Another way to get a more specific model of lesion intensities would be to try to incorporate the cRBM model into the parameter estimation. The constant spatial prior which is now used during the parameter estimation does not allow for pinpointing probable lesion locations inside white matter. Including the more specific cRBM lesion prior into the parameter estimation phase could allow us to learn the GMM parameters for lesions separately from the white matter GMM parameters. This would, furthermore, obviate need for the user-specified intensity-model parameters $\gamma$ and $\pi$, which would make the model more robust. As discussed in the inference section, the cRBM model would complicate the parameter estimation because of the inter-voxel connections. This complication could be overcome using a mean-field approximation.

Despite these difficulties the modeling framework already shows segmentation accuracy on par with state-of-the-art methods. We are quite confident that with the suggested improvements, the segmentation performance of the model can be further increased while also making the model more robust and independent of user-specified parameters. We plan to validate the proposed method on larger data sets of white matter lesions coming from different imaging centers and scanners in order to thoroughly test the contrast-adaptiveness of the approach.

Chapter 5

# Conclusions and contributions

During this PhD project we have developed generative models for segmenting both healthy and pathological brains. We have shown that these types of models can achieve good segmentation performance in both tasks, while being robust to changes in intensity-properties of input scans and able to handle the type of multi-contrast MR data typically used in everyday clinical practice. The adopted generative approach has also two other attractive properties: first, it is a very flexible framework which allows us to extend the models to account for different diseases and abnormalities in the brain. This is achieved by including new prior distributions that capture the properties of the abnormality we aim to model as shown in chapter 4. Second, because the model is generative, we can sample from it which allows us to evaluate how well the model represents the data. This makes for a very interpretable model, where we can see why the model might fail in some cases and why it is successful in others. When using discriminative classifier-based approaches this can be much harder, as the learned parameters might not be directly meaningful in relation to the data, especially when complex classifiers with a large number of parameters are used.

The first part of the project was concentrated on developing a generative model for whole-brain segmentation, which built upon modeling approaches typically used in tissue segmentation. Although we found that the proposed method's segmentation accuracy trailed that of the very best multi-atlas approaches on

an intra-scanner segmentation task, it was shown to be more robust than the benchmark methods when the intensity and resolution properties of the input and training scans do not exactly match. In practice, because manual segmentations are very time consuming to obtain, the latter scenario might be more realistic as we often face situations where the training and target data are acquired on different imaging systems. Due to the parametric modeling approach the method also has a small computational footprint, yielding significantly faster execution times compared to the benchmark tools. Furthermore, the proposed approach was shown to be robust against small training set sizes and able to readily handle multi-contrast data, although for the healthy whole-brain segmentation task, this was shown to not give any significant improvement in terms of segmentation accuracy. Given these properties, our segmentation method would be very useful as an out-of-the-box tool for fast processing of different kinds of MR data sets. Furthermore, because the training labelings are summarized as an atlas which can be shipped along with the software, the potential user does not need to have their own training data in order to do segmentations. When using multi-atlas approaches it can be difficult to get legal permission to distribute the manual labelings, thus requiring that the user has access to training data.

The second part of the project was devoted to including models for MS lesions into the generative framework. The aim was to do simultaneous whole-brain and lesion segmentation in the same contrast-adaptive manner as for the healthy brains. To achieve this, we integrated a novel lesion shape model based on a convolutional restricted Boltzmann machine into the healthy brain segmentation framework by defining new prior and likelihood distributions that account for the presence of lesions. The segmentation accuracy of the proposed model was shown to compare favourably to state-of-the-art methods, while also providing a segmentation of the surrounding neuroanatomy. Having access to not only the lesion segmentations, but also to the segmentations of different brain structures, especially deep gray matter structures such as the thalamus, could be highly valuable for identifying biomarkers related to the progression of MS disease.

### 5.0.1 Other contributions

During the project I have had the chance to be a part of many other fruitful collaborations. Here I list some of the other works that I have contributed to, but which have not been the main focus of the project.

The method for tumor segmentation presented in paper D is based on a very similar framework that was used for lesion segmentation in this project. Here two cRBM models are trained: one to model the full tumor and the other for

modeling the tumor core. Inference is carried out through sampling from the full set of parameters and structure labels, and the final segmentation is obtained using majority voting as described in chapter 4. In this work I closely collaborated with the first author to develop the code to train the cRBM models and to sample from the resulting model, as well as developing the theoretical aspects of the model. Many fruitful discussion about the practical and theoretical problems related to the framework were held also aiding me in my own work. The tumor segmentation model was part of the 2015 multimodal brain tumor segmentation challenge (BRATS) organized in the medical image computing and computer assisted intervention (MICCAI) conference, where it was ranked third best among all methods and first among fully automated segmentation methods. In the future, the plan is to include the mesh-based anatomical prior into the model to yield a joint whole-brain and tumor segmentation framework.

I also collaborated on another tumor segmentation model, presented in paper E, which is based on the currently very popular convolutional neural network (CNN) classifier. Here three 2D CNNs are trained for each orthogonal image plane (axial, sagittal and coronal) and each voxel is then segmented by performing majority-voting on the outputs of the three CNNs. The full segmentation pipeline has three steps: first, the full tumor region is segmented from the background using the CNN outputs, next the segmentation is refined using a cellural-automaton based seed growing method known as grab-cut, finally the different tumor compartments are segmented using again three orthogonal CNNs which were trained to separate the different compartments from the rest of the tumor. The framework was shown to give competitive performance when applied to the MICCAI tumor challenge data. In this work I mainly helped with formulating the segmentation pipeline on a general level and also with writing and proof-reading the paper.

Finally, paper F is a book chapter which gives an overview of the generative segmentation framework for tissue classification. Here my main contribution was making the visualizations which show how segmentation performance improves when the bias field is properly modeled.

Chapter 6

# Future work

In this chapter, we point to possible new avenues for future applications and research based on the models described in the thesis.

## 6.1 Whole brain segmentation

An interesting research direction would be to extend the segmentation framework to include an explicit model of longitudinal data. In the experiments in chapter 3, we used the test-retest data mainly to see if including multi-contrast data would increase the robustness of the segmentations compared to using uni-contrast data only. However, each time point was treated independently although we know that the brain anatomy between the two time points should be fairly similar as all the subjects were healthy. This information could be incorporated explicitly into the model. A longitudinal model would allow us to robustly track the shape and volume changes in different brain structures, which could be used for monitoring disease development in central nervous system disorders such as Alzheimer's or Huntington's disease [RSRF12b]. Longitudinal modeling also in the case of MS disease would be valuable, as clinicians are often interested in how the lesions change over time, and not so much about the actual lesion segmentations. Furthermore, longitudinal tracking of regional atrophy patterns of patients with different MS disease subtypes, such as secondary

progressive MS or relapsing-remitting MS, could give insight into what areas of the brain are mainly affected by the different disease phenotypes [PRG+05].

## 6.2 Lesion segmentation

As an immediate next step, we plan to evaluate the lesion segmentation performance on more data sets, to evaluate the robustness of the method. After that, we will validate the healthy structure segmentations provided by the model on lesioned data. However, in most lesion data sets manual segmentations of the surrounding neuroanatomy are not available, and direct validation using Dice scores can not be done. Thus, we aim to reproduce some known disease effects between MS patients and healthy subjects, such as the atrophy of deep gray matter structures related to MS. The method should be able to differentiate healthy and diseased populations based on the reduction in volume of structures like the thalamus and pallidum. Once the accuracy of the segmentations has been established, we can start trying to exploit the rich set of morphological measures for many different applications such as trying to separate patients with different disease subtypes, or predicting disability scores of individual patients.

Related to disability score prediction, it would likely be beneficial to incorporate patient-specific information beyond the MR data into the segmentation model. This could be done by conditioning the cRBM model on the demographic and clinical data, yielding a model similar to the so-called Bayesian Spatial Generalized Linear Mixed Models (BSGLMM) presented in [GMLB+14]. Here the authors try to predict MS disease subtype of different patients based on the subject-specific manual lesion segmentation and clinical information. In that work only nearest neighbour spatial regularization was used, so it would be interesting to see if the richer spatial model encoded by the cRBM would give increased predictive performance. Furthermore, including the clinical information into the framework would likely also aid the segmentation accuracy as the lesion load and spatial layout of the lesions varies with age and disease subtype.

Chapter 7

# Paper A

# Fast, Sequence Adaptive Parcellation of Brain MR Using Parametric Models

Oula Puonti[1], Juan Eugenio Iglesias[2], and Koen Van Leemput[1,2,3]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark
[2] Martinos Center for Biomedical Imaging, MGH, Harvard Medical School, USA
[3] Departments of Information and Computer Science and of Biomedical Engineering and Computational Science, Aalto University, Finland

**Abstract.** In this paper we propose a method for whole brain parcellation using the type of generative parametric models typically used in tissue classification. Compared to the non-parametric, multi-atlas segmentation techniques that have become popular in recent years, our method obtains state-of-the-art segmentation performance in both cortical and subcortical structures, while retaining all the benefits of generative parametric models, including high computational speed, automatic adaptiveness to changes in image contrast when different scanner platforms and pulse sequences are used, and the ability to handle multi-contrast (vector-valued intensities) MR data. We have validated our method by comparing its segmentations to manual delineations both within and across scanner platforms and pulse sequences, and show preliminary results on multi-contrast test-retest scans, demonstrating the feasibility of the approach.

## 1 Introduction

Computational methods for automatically segmenting magnetic resonance (MR) images of the brain have seen tremendous advances in recent years. So-called *tissue classification* methods, which aim at extracting the white matter, gray matter, and cerebrospinal fluid, are now well established. In their simplest form, these methods classify voxels independently based on their intensity alone, although state-of-the-art methods often incorporate a probabilistic atlas – a parametric representation of prior neuroanatomical knowledge that is learned from manually annotated training data – as well as explicit models of MR imaging artifacts [1–3]. Tissue classification techniques have a number of attractive properties, including their computational speed and their ability to automatically adapt to changes in image contrast when different scanner platforms and pulse sequences are used. Furthermore, they can readily handle the multi-contrast (vector-valued intensities) MR scans that are acquired in clinical imaging, and can include models of pathology such as white matter lesions and brain tumors.

Despite these strengths, attempts at expanding the scope of tissue classification techniques to also segment dozens of subcortical structures have been less successful [4]. In that area, better results have been obtained with so-called

K. Mori et al. (Eds.): MICCAI 2013, Part I, LNCS 8149, pp. 727–734, 2013.
© Springer-Verlag Berlin Heidelberg 2013

*multi-atlas* techniques – non-parametric methods in which a collection of manually annotated images are deformed onto the target image using pair-wise registration, and the resulting atlases are fused to obtain a final segmentation [4, 5]. Although early methods used a simple majority voting rule, recent developments have concentrated on exploiting local intensity information to guide the atlas fusion process, which is particularly helpful in cortical areas for which accurate inter-subject registration is challenging [6, 7].

Although multi-atlas techniques have been shown to provide excellent segmentation results, they do come with a number of distinct disadvantages compared to tissue classification techniques. Specifically, their non-parametric nature entails a significant computational burden because of the large number of pair-wise registrations that is required for each new segmentation. Furthermore, their applicability across scanner platforms and pulse sequences is seldom addressed, and it remains unclear how multi-contrast MR and especially pathology can be handled with these methods.

In this paper, we revisit tissue classification modeling techniques and demonstrate that it is possible to obtain cortical and subcortical segmentation accuracies that are on par with the current state-of-the-art in multi-atlas segmentation, while being dramatically faster. Following a modeling approach similar to [1, 3] for tissue classification, but with a carefully computed probabilistic atlas of 41 brain substructures, we show excellent performance both within and across scanner platforms and pulse sequences. Compared to other methods aiming at sequence adaptive whole brain segmentation, we do not require specific MR sequences for which a physical forward model is available [8], and we segment many more structures without a priori defined contrast-specific initializations as in [2].

## 2 Modeling Framework

We use a Bayesian modeling approach, in which a generative probabilistic image model is constructed and subsequently "inverted" to obtain automated segmentations. We first describe our generative model, and subsequently explain how we use it to obtain automated segmentations. Because of space constraints, we only describe the uni-contrast case here (i.e., a scalar intensity value for each voxel); the generalization to multi-contrast data is straightforward [3].

### 2.1 Generative Model

Our model consists of a prior distribution that predicts where anatomical labels typically occur throughout brain images, and a likelihood distribution that links the resulting labels to MR intensities. As a segmentation prior we use a recently proposed tetrahedral mesh-based probabilistic atlas [9], where each mesh node contains a probability vector containing the probabilities for the $K$ different brain structures under consideration. The resolution of the mesh is locally adaptive, being sparse in large uniform regions and dense around the structure borders.

The positions of the mesh nodes, denoted by $\boldsymbol{\theta}_l$, can move according to a deformation prior $p(\boldsymbol{\theta}_l)$ that prevents the mesh from tearing or folding onto itself. The prior probability of label $l_i \in \{1, ..., K\}$ in voxel $i$ is denoted by $p_i(l_i|\boldsymbol{\theta}_l)$, which is computed by interpolating the probability vectors in the vertices of the deformed mesh. Assuming conditional independence of the labels between voxels given the mesh node positions, the prior probability of a segmentation is then given by $p(\mathbf{l}|\boldsymbol{\theta}_l) = \prod_i^I p_i(l_i|\boldsymbol{\theta}_l)$, where $\mathbf{l} = (l_1, ..., l_I)^T$ denotes a complete segmentation of an image with $I$ voxels.

For the likelihood distribution, we associate a mixture of Gaussian distributions with each neuroanatomical label to model the relationship between segmentation labels and image intensities [1]. To account for the smoothly varying intensity inhomogeneities that typically corrupt MR scans, we model such bias fields as a linear combination of spatially smooth basis functions [3]. Letting $\mathbf{d} = (d_1, ..., d_I)^T$ denote a vector containing the image intensities in all voxels, and $\boldsymbol{\theta}_d$ a vector collecting all bias field and Gaussian mixture parameters, the likelihood distribution then takes the form $p(\mathbf{d}|\mathbf{l}, \boldsymbol{\theta}_d) = \prod_{i=1}^I p_i(d_i|l_i, \boldsymbol{\theta}_d)$, where

$$p_i(d|l, \boldsymbol{\theta}_d) = \sum_{g=1}^{G_l} \mathcal{N}\Big(d - \sum_{p=1}^{P} c_p \phi_p^i \,\Big|\, \mu_{lg}, \sigma_{lg}^2\Big) w_{lg}$$

and $\mathcal{N}\big(d|\mu, \sigma^2\big) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(d-\mu)^2}{2\sigma^2}\Big)$. Here $G_l$ is the number of Gaussian distributions associated with label $l$; and $\mu_{lg}$, $\sigma_{lg}^2$, and $w_{lg}$ are the mean, variance, and weight of component $g$ in the mixture model of label $l$. Furthermore, $P$ denotes the number of bias field basis functions, $\phi_p^i$ is the basis function $p$ evaluated at voxel $i$, and $c_p$ its coefficient. To complete the model, we assume a flat prior on $\boldsymbol{\theta}_d$: $p(\boldsymbol{\theta}_d) \propto 1$.

### 2.2 Inference

Using the model described above, the most probable segmentation for a given MR scan is obtained as $\hat{\mathbf{l}} = \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{d}) = \arg\max_{\mathbf{l}} \int p(\mathbf{l}|\mathbf{d}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{d}) \mathrm{d}\boldsymbol{\theta}$, where $\boldsymbol{\theta} = (\boldsymbol{\theta}_d, \boldsymbol{\theta}_l)^T$ collects all the model parameters. This requires an integration over all possible parameter values, each weighed according to its posterior $p(\boldsymbol{\theta}|\mathbf{d})$. Since this integration is intractable we approximate it by estimating the parameters with maximum weight $\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{d})$, and using the contribution of those parameters only:

$$\hat{\mathbf{l}} = \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{d}) \approx \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{d}, \hat{\boldsymbol{\theta}}) = \arg \max_{\{l_1, ... l_I\}} \prod_{i=1}^{I} p_i(l_i|d_i, \hat{\boldsymbol{\theta}}). \tag{1}$$

The optimization of eq. (1) is tractable because it involves maximizing $p_i(l_i|d_i, \hat{\boldsymbol{\theta}}) \propto p_i(d_i|l_i, \hat{\boldsymbol{\theta}}_d) p_i(l_i|\hat{\boldsymbol{\theta}}_l)$ in each voxel independently.

To find the optimal parameters we maximize

$$p(\boldsymbol{\theta}|\mathbf{d}) \propto p(\mathbf{d}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \propto \left( \prod_{i=1}^{I} \sum_{l=1}^{K} p_i(d_i|l, \boldsymbol{\theta}_d) p_i(l|\boldsymbol{\theta}_l) \right) p(\boldsymbol{\theta}_l) \tag{2}$$

by iteratively keeping the mesh positions $\boldsymbol{\theta}_l$ fixed at their current values and updating the remaining parameters $\boldsymbol{\theta}_d$, and vice versa, until convergence. For the mesh node position optimization we use a a standard conjugate gradient optimizer, and for the remaining parameters a dedicated generalized expectation-maximization (GEM) algorithm similar to [3]. In particular, the GEM optimization involves iteratively computing the following "soft" assignments in all voxels $i \in \{1, \ldots, I\}$:

$$q_i^{lg} = \frac{w_{lg}\mathcal{N}\big(d_i - \sum_{p=1}^{P} c_p \phi_p^i \,\big|\mu_{lg}, \sigma_{lg}^2\big) p_i(l|\boldsymbol{\theta}_l)}{\sum_{k=1}^{K} p_i(d_i|k, \boldsymbol{\theta}_d) p_i(k|\boldsymbol{\theta}_l)}, \; \forall l \in \{1, \ldots, K\}, \forall g \in \{1, \ldots, G_l\}$$

based on the current parameter estimates, and subsequently updating the parameters accordingly:

$$\mu_{lg} \leftarrow \frac{\sum_{i=1}^{I} q_i^{lg} \Big(d_i - \sum_{p=1}^{P} c_p \phi_p^i\Big)}{\sum_{i=1}^{I} q_i^{lg}}, \;\; \sigma_{lg}^2 \leftarrow \frac{\sum_{i=1}^{I} q_i^{lg} \Big(d_i - \mu_{lg} - \sum_{p=1}^{P} c_p \phi_p^i\Big)^2}{\sum_{i=1}^{I} q_i^{lg}},$$

$$w_{lg} \leftarrow \frac{\sum_{i=1}^{I} q_i^{lg}}{\sum_{g=1}^{G_l} \sum_{i=1}^{I} q_i^{lg}}, \quad (c_1, \ldots, c_P)^T \leftarrow \left(\mathbf{A}^T \mathbf{S} \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{S} \mathbf{r},$$

where

$$\mathbf{A} = \begin{pmatrix} \phi_1^1 & \ldots & \phi_P^1 \\ \vdots & \ddots & \vdots \\ \phi_1^I & \ldots & \phi_P^I \end{pmatrix}, \quad \mathbf{S} = diag(s_i), \quad s_i = \sum_{l=1}^{K} \sum_{g=1}^{G_l} \frac{q_i^{lg}}{\sigma_{lg}^2},$$

$$\mathbf{r} = (r_1, .., r_I)^T, \quad r_i = d_i - \tilde{d}_i, \quad \tilde{d}_i = \frac{\sum_{l=1}^{K} \sum_{g=1}^{G_l} s_i^{lg} \mu_{lg}}{\sum_{l=1}^{K} \sum_{g=1}^{G_l} s_i^{lg}}.$$

## 3 Implementation

We used a training dataset of 39 T1-weighted scans and corresponding expert delineations of 41 brain structures to build our mesh-based atlas and to run pilot experiments to tune the settings of our algorithm. The scans were acquired on a 1.5T Siemens Vision scanner using a magnetisation prepared, rapid acquisition gradient-echo (MPRAGE) sequence (voxel size $1.0 \times 1.0 \times 1.0$ mm$^3$). The 39 subjects are a mix of young, middle-aged, and old healthy subjects, as well as patients with either questionable or probable Alzheimer's disease [6].

We used 15 randomly picked subjects out of the available 39 to build our probabilistic atlas. The remaining subjects were used to find suitable settings for our algorithm. After experimenting, we decided to restrict sub-structures with similar intensity properties to having the same GMM parameters, e.g., left and right hemisphere white matter are modeled as having the same intensity properties. Further, we experimentally set a suitable value for the number of Gaussians for each label (variable $G_l$): three for gray matter structures, cerebro-spinal fluid,

and non-brain tissues; and two for white matter structures, thalamus, putamen, and pallidum.

To initialize the algorithm, we co-register our atlas to the target image using an affine transformation. For this purpose we use the method described in [10], which uses atlas probabilities, rather than an intensity template, to drive the registration process. As is common in the literature, the MR intensities are log-transformed because of the additive bias field model that is employed [3].

## 4 Experiments

To validate the proposed algorithm, we performed experiments on two datasets of T1-weighted images that were manually labeled using the same protocol as the training data, each acquired on a different scanner platform and with a different pulse sequence. We also show preliminary results on a third dataset that consists of test-retest scans of multi-contrast (T1- and T2-weighted) images without manual annotations. We emphasize that we ran our method on all three datasets using the exact same settings.

Although our method segments 41 structures in total, some of the structures are not typically validated (e.g., left/right choroid plexus, left/right vessels), thus we here report quantitative results for a subset of 23 structures: cerebral white matter (WM), cerebellum white matter (CWM), cerebral cortex (CT), cerebellum cortex (CCT), lateral ventricle (LV), hippocampus (HP), thalamus (TH), caudate (CA), putamen (PU), pallidum (PA) and amygdala (AM), for both the left and the right side, along with brainstem (BS). In order to gauge the performance of our method with respect to the current state-of-the-art in the field, we also report results for the well-known FreeSurfer package [11] and two multi-atlas segmentation methods: BrainFuse [6], which uses a Gaussian kernel to perform local intensity-based atlas weighing, and Majority Voting [5], which weighs each atlas equally. We note that all three competing methods used the same training data described in section 3, ensuring a fair comparison: FreeSurfer to build its label and intensity models; and the multi-atlas methods to perform the pair-wise atlas propagations and to tune optimal parameter settings. All three competing methods apply the same preprocessing stages to skull-strip the images, remove bias field artifacts, and perform intensity normalization as described in [11]. The proposed method works directly on the input data itself without preprocessing. For our implementation of Majority Voting, we used the pair-wise registrations computed by BrainFuse.

Figure 1(a) shows the Dice scores (averaged across left and right) between the automated and manual segmentations for the four methods on our first dataset, which consists of T1-weighted images of 13 subjects acquired with the same Siemens scanner and MPRAGE pulse sequence as the training data. Note that FreeSurfer, BrainFuse, and Majority Voting are specifically trained for this type of data, whereas the proposed method is not. It can be seen that each method gives quite accurate and comparable segmentations, except for majority voting, which clearly trails the other methods. The mean Dice score across these structures is 0.859 for the suggested method, 0.864 for BrainFuse, 0.853 for FreeSurfer,

and 0.793 for Majority Voting. Table 1 shows the execution times for the methods. The experiments were run on a cluster where each node has two quad-core Xeon 5472 3.0GHz CPUs and 32GB of RAM. Only one core was used for the experiments. The multi-atlas methods require computationally heavy pair-wise registrations and thus have the longest run times, followed by FreeSurfer which is somewhat faster. The suggested method is clearly the fastest of the four, being approximately 26 times faster than BrainFuse or Majority Voting and 15 times faster than FreeSurfer. We note that our method is implemented using Matlab with the atlas deformation parts wrapped in C++, and in no way optimized for speed. To conclude our experiments on this dataset, table 2 shows how the number of training subjects in the atlas affects the mean segmentation accuracy of the proposed method, indicating that the method benefits from the availability of more training data.

Figure 1(b) shows the Dice scores on our second dataset, which consists of 14 T1-weighted MR scans that were acquired with a 1.5T GE Signa scanner using a spoiled gradient recalled (SPGR) sequence (voxel size $0.9375 \times 0.9375 \times 1.5$ mm$^3$). The overall segmentation accuracy of each method is decreased compared to the Siemens data, which is likely due to poorer image contrast as a result of the different pulse sequence and a slightly lower image resolution. Both FreeSurfer and our method are able to sustain an overall accuracy of 0.798, while the accuracies of BrainFuse and Majority Voting decrease to 0.746 and 0.70 respectively. The relatively good performance of FreeSurfer, which is trained specifically on the Siemens image contrast, can be explained by its in-built renormalization procedure for T1 acquisitions, which applies a multi-linear atlas-image registration and a histogram matching step to update the class-conditional densities for each structure [12]. The multi-atlas methods, in contrast, directly incorporate the Siemens contrast in the segmentation process, and would likely benefit from a retuning of their parameters for this specific application. Note that the proposed method requires no renormalization or retuning to perform well.

As a preliminary demonstration of the multi-contrast segmentation abilities of our method, figure 1(c) shows a measurement of volume differences between both uni-contrast (T1) and multi-contrast (T1 + T2) repeat scans of five individuals. For each subject, a multi-contrast scan was acquired with an identical Siemens 3T Tim Trio scanner at two different facilities, with a interval between the two scan sessions of maximum 3 months. The scans consist of a very fast (under 5 min total acquisition time) T1-weighted and bandwidth-matched T2-weighted image (multi-echo MPRAGE sequence for T1 and 3D T2-SPACE sequence for T2, voxel size $1.2 \times 1.2 \times 1.2$ mm$^3$). The volume difference in a structure was computed as the absolute difference between the volumes estimated at the two time points, normalized by the average of the volumes, both when only the T1-weighted image was used, and when T1 and T2 were used. The figure shows that our method seems to work as well on multi-contrast as on uni-contrast data, opening possibilities for simultaneous brain lesion segmentation in the future. An example segmentation of one of the multi-contrast scans is shown in figure 2.

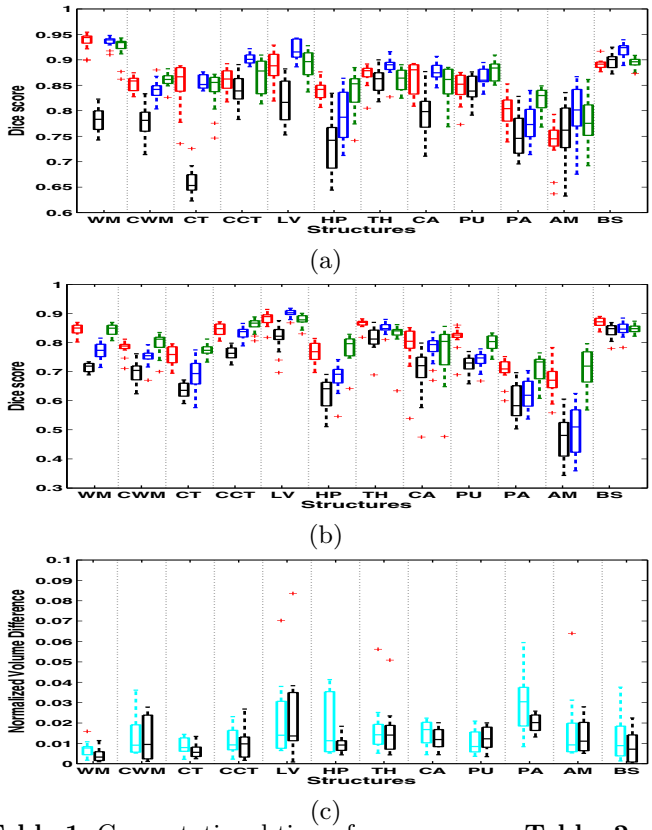



**Fig. 1.** (a) Dice scores of the first dataset (Siemens). FreeSurfer is red, BrainFuse blue, Majority Voting black, and the suggested method green. (b) Dice scores of the second data set (GE). (c) Normalized volume differences: multi-contrast data is cyan, and T1-only black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'.

**Table 1.** Computational times for the four different methods

| Method | Comp. time(h) |
|---|---|
| BrainFuse | $\sim$ 17 |
| Majority voting | $\sim$ 16 |
| FreeSurfer | $\sim$ 9.5 |
| Suggested method | $\sim$ 0.6 |

**Table 2.** Average Dice score across all structures for the first (Siemens) dataset vs. number of training subjects

| Number of subjects | Mean Dice score |
|---|---|
| 5 | 0.820 |
| 9 | 0.843 |
| 15 | 0.859 |

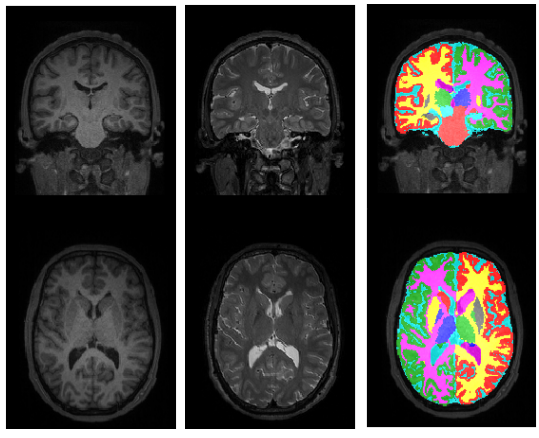


**Fig. 2.** An example of a multi-contrast segmentation generated by the proposed method

## 5 Discussion

In this paper we proposed a method for whole brain parcellation using the type of generative parametric models typically used in tissue classification techniques. Comparisons with current state-of-the-art methods demonstrated excellent performance both within and across scanner platforms and pulse sequences, as well as a large computational advantage. Future work will concentrate on a more thorough validation of the method's multi-contrast segmentation performance. We also plan to use other validation metrics beyond the mere spatial overlap used in this paper, such as volumetric and boundary distance measures.

**Acknowledgements.** This research was supported by NIH NCRR (P41-RR14075), NIBIB (R01EB013565), Academy of Finland (133611), TEKES (CombBrain), and financial contributions from the Technical University of Denmark.

## References

1. Ashburner, J., Friston, K.: Unified segmentation. Neuroimage 26, 839–885 (2005)
2. Bazin, P.L., Pham, D.L.: Homeomorphic brain image segmentation with topological and statistical atlases. Medical Image Analysis 12(5), 616–625 (2008)
3. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based bias field correction of MR images of the brain. IEEE Transactions on Medical Imaging 18(10), 885–896 (1999)
4. Babalola, K.O., Patenaude, B., Aljabar, P., Schnabel, J., Kennedy, D., Crum, W., Smith, S., Cootes, T., Jenkinson, M., Rueckert, D.: An evaluation of four automatic methods of segmenting the subcortical structures in the brain. Neuroimage 47(4), 1435–1447 (2009)
5. Heckemann, R., Hajnal, J., Aljabar, P., Rueckert, D., Hammers, A.: Automatic anatomical brain MRI segmentation combining label propagation and decision fusion. Neuroimage 33, 115–126 (2006)
6. Sabuncu, M.R., Yeo, B., Van Leemput, K., Fischl, B., Golland, P.: A generative model for image segmentation based on label fusion. IEEE Transactions on Medical Imaging 29(10), 1714–1729 (2010)
7. Ledig, C., Wolz, R., Aljabar, P., Lotjonen, J., Heckemann, R.A., Hammers, A., Rueckert, D.: Multi-class brain segmentation using atlas propagation and EM-based refinement. In: 9th IEEE International Symposium on Biomedical Imaging (ISBI), pp. 896–899 (2012)
8. Fischl, B., Salat, D.H., van der Kouwe, A.J., Makris, N., Ségonne, F., Quinn, B.T., Dale, A.M.: Sequence-independent segmentation of magnetic resonance images. Neuroimage 23, S69–S84 (2004)
9. Van Leemput, K.: Encoding probabilistic brain atlases using Bayesian inference. IEEE Transactions on Medical Imaging 28(6), 822–837 (2009)
10. D'Agostino, E., Maes, F., Vandermeulen, D., Suetens, P.: Non-rigid atlas-to-image registration by minimization of class-conditional image entropy. In: Barillot, C., Haynor, D.R., Hellier, P. (eds.) MICCAI 2004. LNCS, vol. 3216, pp. 745–753. Springer, Heidelberg (2004)
11. Dale, A., Fischl, B., Sereno, M.: Cortical surface-based analysis I: Segmentation and surface reconstruction. Neuroimage 9, 179–194 (1999)
12. Han, X., Fischl, B.: Atlas renormalization for improved brain MR image segmentation across scanner platforms. IEEE Transactions on Medical Imaging 26(4), 479–486 (2007)

Chapter 8

# Paper B

# Fast and Sequence-Adaptive Whole-Brain Segmentation Using Parametric Bayesian Modeling

Oula Puonti[a,*], Juan Eugenio Iglesias[b,c], Koen Van Leemput[a,c]

[a] *Department of Applied Mathematics and Computer Science, Technical University of Denmark, Richard Petersens Plads, Building 321 DK-2800 Kgs. Lyngby, Denmark*

[b] *Basque Center on Cognition, Brain and Language (BCBL), Paseo Mikeletegi, 20009 San Sebastian - Donostia, Gipuzkoa, Spain*

[c] *Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Harvard Medical School, 149 13th St, Charlestown, MA 02129, USA*

**Abstract**

Quantitative analysis of magnetic resonance imaging (MRI) scans of the brain requires accurate automated segmentation of anatomical structures. A desirable feature for such segmentation methods – especially when they are publicly released – is to be robust against changes in acquisition platform and imaging protocol. In this paper we validate the performance of a segmentation algorithm designed to meet these requirements, building upon generative parametric models previously used in tissue classification. The method is tested on four different datasets acquired with different scanners and pulse sequences, demonstrating comparable accuracy to state-of-the-art methods on T1-weighted scans while being one to two orders of magnitude faster. The proposed algorithm is shown to be robust against small training datasets, and readily handles images with different MRI contrast as well as multi-contrast data. The software used in this article is publicly available and can be downloaded from the Neuroimaging Informatics Tools and Resources Clearinghouse (`http://www.nitrc.org`)[1].

*Keywords:* MRI, Segmentation, Atlases, Parametric modeling, Unsupervised modeling

## 1. Introduction

So-called *whole-brain segmentation* techniques aim to automatically label a multitude of cortical and subcortical regions from brain MRI scans. Recent years have seen tremendous advances in this field, enabling, for the first time, fine-grained comparisons of regional brain morphometry between large groups of subjects. Current state-of-the-art whole-brain segmentation algorithms are typically based on supervised models of image appearance in T1-weighted scans, in which the relationship between intensities and neuroanatomical labels is learned from a set of manually annotated training images.

This approach suffers from two fundamental limitations. First, segmentation performance often degrades when the algorithms are applied to T1-weighted data acquired on different scanner platforms or using different imaging sequences, due to subtle changes in the obtained image contrast (Han and Fischl, 2007; Roy et al., 2013). And second, the exclusive focus on only T1-weighted images hinders the ultimate translation of whole-brain segmentation techniques into clinical practice, where they hold great potential to support personalized treatment of patients suffering from brain disease. This is because clinical imaging uses additional MRI contrast mechanisms to show clinically relevant information, including T2-weighted or fluid attenuated inversion recovery (FLAIR) images that are much more sensitive to certain pathologies than T1-weighted scans (e.g., white matter lesions or brain tumors). Although incorporating models of lesions into whole-brain segmentation techniques is an open problem in itself, a first necessary step towards bringing these techniques into clinical practice is to make them capable of handling the multi-contrast images that are acquired in standard clinical routine.

In this article, we present and validate the performance of a fast, sequence-independent whole-brain segmentation algorithm. The method, which is based on a mesh-based computational atlas combined with a Gaussian appearance model, yields segmentation accuracies

*Corresponding author, email: oupu@dtu.dk

[1] We will upload the software to NITRC when this work is published.

comparable to the state-of-the-art; automatically adapts to different MRI contrasts (even if multimodal); requires only a small amount of training data; and achieves computational times comparable to those of the fastest algorithms in the field (Zikic et al., 2014; Ta et al., 2014).

### *1.1. Current state-of-the-art in whole-brain segmentation*

Early methods for the segmentation of brain structures often relied on *parametric* models, in which the available training data were summarized in relevant statistics that were subsequently used to inform the segmentation of previously unseen subjects. Because many distinct brain structures have similar intensity characteristics in MRI, these methods were typically built around detailed probabilistic models of the expected shape and relative positioning of different brain regions, using surface-based (Kelemen et al., 1998; Pizer et al., 2003; Patenaude et al., 2011; Cootes et al., 1998) or volumetric (Fischl et al., 2002; Pohl et al., 2006b) models. These anatomical models were then combined with supervised models of appearance to encode the typical intensity characteristics of the relevant structures in the training data, often using Gaussian models for either the intensity of individual voxels (Fischl et al., 2002; Pohl et al., 2006b) or for entire regional intensity profiles (Kelemen et al., 1998; Pizer et al., 2003; Patenaude et al., 2011; Cootes et al., 1998). The segmentation problem was then formulated in a Bayesian setting, in which segmentations were sought that satisfy both the shape and appearance constraints.

More recently, *non-parametric* methods have gained increasing attention in the field of whole-brain segmentation, mostly in the form of multi-atlas label fusion (Rohfling et al., 2004a; Heckemann et al., 2006; Isgum et al., 2009; Artaechevarria et al., 2009; Sabuncu et al., 2010; Rohfling et al., 2004b; Wang et al., 2013; Coupé et al., 2011; Rousseau et al., 2011; Tong et al., 2013; Wu et al., 2013; Asman and Landman, 2013; Zikic et al., 2014). In these methods, each of the manually annotated training scans is first deformed onto the target image using an image registration algorithm. Then, the resulting deformation fields are used to warp the manual annotations, which are subsequently fused into a final consensus segmentation. Although early methods used a simple majority voting rule (Rohfling et al., 2004a; Heckemann et al., 2006), recent developments have concentrated on exploiting local intensity information to guide the atlas fusion process. This is particularly helpful in cortical areas, for which accurate inter-subject registration is challenging (Sabuncu et al., 2010; Ledig et al., 2012b). Label fusion methods have been shown to yield very accurate whole-brain segmentations (Landman and Warfield, 2012), but their accuracy comes at the expense of a high computational cost as a result of the multiple non-linear registrations that are required. Efforts to alleviate this issue include a local search using entire image patches, such that much faster *linear* registrations can be used (Coupé et al., 2011; Ta et al., 2014), as well as using rich contextual features so that only a single non-linear warp is needed (Zikic et al., 2014).

### *1.2. Existing methods that handle changes in MRI contrast*

Since both the parametric and non-parametric methods reviewed above are *supervised*, they explicitly encode the specific image contrast properties of the dataset used for training. This poses limitations on their ability to segment images that were acquired with different scanners or imaging sequences than the training scans.

A generic way of making supervised whole-brain segmentation methods work across imaging platforms is histogram matching (also known as intensity normalization), in which the intensity profiles of new images are altered so as to resemble those of the images used for training (Nyúl et al., 2000; Roy et al., 2013). However, histogram matching can only be used when the training and target data have been acquired with the same type of MRI sequence (e.g., T1-weighted), and it does not completely cancel the negative effects that intensity mismatches have on segmentation accuracy (Roy et al., 2013).

Another approach is to have the training dataset include images that are representative of all the scanners and protocols that are expected to be encountered in practice. However, this approach quickly becomes impractical due to the large number of possible combinations of MRI hardware and acquisition parameters. The situation is exacerbated for clinical data, due to the lack of standardized protocols to acquire multi-contrast MRI data across clinical imaging centers.

In contrast synthesis (Roy et al., 2013), the original scan is not directly segmented, but rather used to generate a new scan with the desired intensity profile, which is then segmented instead. The premise of this technique is that a database of scans acquired with both the source and target contrast is available, so that the relationship between the two can be learned (Iglesias et al., 2013a; Roy et al., 2013). This approach makes it unnecessary to manually annotate additional training data for each new set-up that is considered – a considerable advantage given that a manual whole-brain segmentation often takes several days per scan (Fischl et al., 2002).

However, it still requires that additional example subjects are scanned with both the source and target scanner and protocol, which is not always practical.

Finally, a more fundamental way to address the problem is to perform whole-brain segmentation in the space of intrinsic MRI tissue parameters (Fischl et al., 2004b). However, this requires the usage of specific MRI sequences for which a physical forward model is available, which are not widely implemented on MRI scanning platforms, and particularly not on clinical systems.

### *1.3. Contribution: a fast, sequence-adaptive whole-brain segmentation algorithm*

In contrast to the aforementioned approaches to whole-brain segmentation, which rely on supervised models of the specific intensity profiles seen in the training data, in this paper we advocate an *unsupervised* approach that automatically learns appropriate intensity models from the images being analyzed. At the core of the method is an intensity clustering algorithm (a Gaussian mixture model) that derives its independence of specific image contrast properties by simply grouping together voxels with similar intensities. This approach is well-established for the purpose of tissue classification (aimed at extracting the white matter, gray matter and cerebrospinal fluid) where it is typically augmented with models of MRI imaging artifacts (Wells et al., 1996; Van Leemput et al., 1999a; Ashburner and Friston, 2005) and spatial models such as probabilistic atlases (Ashburner and Friston, 1997; Van Leemput et al., 1999a; Ashburner and Friston, 2005) or Markov random fields (Van Leemput et al., 1999b; Zhang et al., 2001). In this paper, we build on these techniques, using a mesh-based probabilistic atlas that provides whole-brain segmentation accuracy at the level of the state-of-the-art, both within and across scanner platforms and pulse sequences. Unlike many other techniques, the method does not need any pre-processing such as skull stripping, bias field correction or intensity normalization. Furthermore, because the method is parametric, only a single non-linear registration (of the atlas to the target image) is required, yielding a very fast overall computational footprint.

*Related work.* Since the method we propose combines Gaussian mixture modeling with MRI bias field correction and probabilistic atlas deformation, it is closely related to the unified segmentation framework described in (Ashburner and Friston, 2005); however only basic tissue classification on T1-weighted images was attempted in that work. A related method based on fuzzy c-means clustering and a topological atlas was described in (Bazin and Pham, 2008), but that only segmented a handful of structures, and relied on the availability of pre-defined centroid initializations for each type of MRI sequence the method is expected to encounter.

An early attempt at whole-brain segmentation using a deformable probabilistic atlas combined with unsupervised intensity clustering was described in (Babalola et al., 2009); however, the atlas registration was performed independently of the segmentation process, using relatively coarse deformations, and the resulting segmentation performance was found to trail that of label fusion methods. Subsequent methods showing better performance (Ledig et al., 2012a, 2015; Makropoulos et al., 2014; Iglesias et al., 2013b) used the nonparametric paradigm instead, where a probabilistic atlas is computed in the space of the target scan, i.e., after warping each of the training scans onto the target image using pairwise registration. Such approaches are computationally much more expensive than the parametric method we advocate here.

An early version of this work, along with a preliminary validation, was presented in (Puonti et al., 2013). The current article adds a more detailed explanation of our modeling approach, quantitative comparisons with additional state-of-the-art label fusion algorithms, and a more extensive validation – particularly regarding test-retest reliability, segmentation of multi-contrast data, and the sensitivity of the method to the size of the training dataset.

## 2. Modeling framework

Let $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_I)$ denote a matrix collecting the intensities in a multi-contrast brain MRI scan with $I$ voxels, where the vector $\mathbf{d}_i = (d_i^1, \ldots, d_i^N)^T$ contains the intensities in voxel $i$ for each of the available $N$ contrasts. Furthermore, let $\mathbf{l} = (l_1, \ldots, l_I)$ be the corresponding segmentation, where $l_i \in \{1, \ldots, K\}$ denotes the one of $K$ possible segmentation labels assigned to voxel $i$.

In order to estimate $\mathbf{l}$ from $\mathbf{D}$, i.e., to compute automated segmentations, we use a generative modeling approach: a forward probabilistic model of MRI images is defined, and subsequently "inverted" to obtain the segmentation. The model consists of two parts: a prior and a likelihood. The prior is a probability distribution over segmentations $p(\mathbf{l})$ that encodes prior knowledge on human neuroanatomy. The likelihood is a probability distribution over image intensities that is conditioned on the segmentation $p(\mathbf{D}|\mathbf{l})$, which models the imaging process through which a certain segmentation yields the observed MRI scan. This type of model is generative be-

cause it provides a mechanism to generate data through the forward model: in our case, we could generate a random brain MRI scan by first sampling the prior to obtain a segmentation, and then sampling the likelihood conditioned on the resulting segmentation.

Within this framework, the posterior distribution of image segmentations given an input brain MRI scans is given by Bayes' rule:

$$p\left(\mathbf{l}|\mathbf{D}\right) \propto p\left(\mathbf{D}|\mathbf{l}\right) p\left(\mathbf{l}\right). \tag{1}$$

Maximizing Eq. 1 with respect to $\mathbf{l}$ then yields the maximum a posteriori (MAP) estimate of the segmentation.

In the rest of this Section, we will describe in depth the prior (Section 2.1) and likelihood (Section 2.2); we will propose an inference algorithm to approximately maximize Eq. 1 (Section 2.3); and finally we will describe the details of the implementation of this algorithm (Section 2.4).

### 2.1. Prior

For the prior $p(\mathbf{l})$ we use a generalization of the probabilistic brain atlases often used in brain MRI segmentation (Ashburner and Friston, 1997; Van Leemput et al., 1999b,a, 2001; Zijdenbos et al., 2002; Fischl et al., 2002; Ashburner and Friston, 2005; Prastawa et al., 2005; Pohl et al., 2006b; D'Agostino et al., 2006; Awate et al., 2006; Pohl et al., 2007). This model, detailed in (Van Leemput, 2009), is based on a deformable tetrahedral mesh, the properties of which are learned automatically from a set of manual example segmentations made on MRI scans of training subjects. Each of the vertices of the mesh has an associated set of label probabilities specifying how frequently each of the $K$ labels occurs at the vertex. The resolution of the mesh is locally adaptive, being sparse in large uniform regions and dense around the structure borders. This automatically introduces a locally varying amount of spatial blurring in the resulting atlas, aiming to avoid over-fitting of the model to the available training samples (Van Leemput, 2009). During training, the topology of the mesh and the position of its vertices in atlas space (henceforth "reference position") is computed along with the label probabilities in a non-linear, group-wise registration of the labeled training data. An example of the resulting probabilistic brain atlas, computed from manual parcellations in 20 subjects, is displayed in its reference position in Figure 1; note the irregularity in the shapes and sizes of the tetrahedra.

The positions of the mesh nodes $\mathbf{x}$ can change according to their prior distribution $p(\mathbf{x})$:

$$p(\mathbf{x}) \propto \exp\left(-\beta \sum_{t=1}^{T} \phi_t(\mathbf{x}, \mathbf{x}_{ref})\right) \tag{2}$$

where $T$ and $\mathbf{x}_{ref}$ denote the number of tetrahedra and the reference position of the mesh, respectively; $\phi_t(\mathbf{x}, \mathbf{x}_{ref})$ is a penalty for deforming tetrahedron $t$ from its reference to its actual position; and $\beta > 0$ is a scalar that controls the global stiffness of the mesh. We use the penalty term proposed in (Ashburner et al., 2000), which goes to infinity when the Jacobian determinant of the deformation approaches zero. This choice prevents the mesh from tearing or folding onto itself, thus preserving its topology.

Given a deformed mesh with node positions $\mathbf{x}$, the probability $p_i(k|\mathbf{x})$ of observing label $k$ at a voxel $i$ is obtained by barycentric interpolation of the label probabilities at the vertices of the tetrahedron containing the voxel. Moreover, we assume conditional independence of the labels of the different voxels given the mesh node positions, such that

$$p(\mathbf{l}|\mathbf{x}) = \prod_{i=1}^{I} p_i(l_i|\mathbf{x}). \tag{3}$$

The expression for the prior distribution over segmentations is finally:

$$p(\mathbf{l}) = \int_{\mathbf{x}} p(\mathbf{l}|\mathbf{x}) p(\mathbf{x}) \mathrm{d}\mathbf{x}. \tag{4}$$

### 2.2. Likelihood

The likelihood $p(\mathbf{D}|\mathbf{l})$ models the relationship between segmentation labels and image intensities. For this purpose, we associate a mixture of Gaussian distributions with each label (Ashburner and Friston, 2005), and assume that the bias field imaging artifact typically seen in MRI can be modeled as a multiplicative and spatially smooth effect (Wells et al., 1996). For computational reasons, we use log-transformed image intensities in $\mathbf{D}$, and model the bias field as a linear combination of spatially smooth basis functions that is *added* to the local voxel intensities (Van Leemput et al., 1999a).

Specifically, letting $\boldsymbol{\theta}$ denote all bias field and Gaussian mixture parameters, with uniform prior $p(\boldsymbol{\theta}) \propto 1$, the likelihood is defined by

$$p(\mathbf{D}|\mathbf{l}) = \int_{\boldsymbol{\theta}} p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}, \tag{5}$$

where

$$p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}) = \prod_{i=1}^{I} p_i(\mathbf{d}_i|l_i, \boldsymbol{\theta}), \tag{6}$$

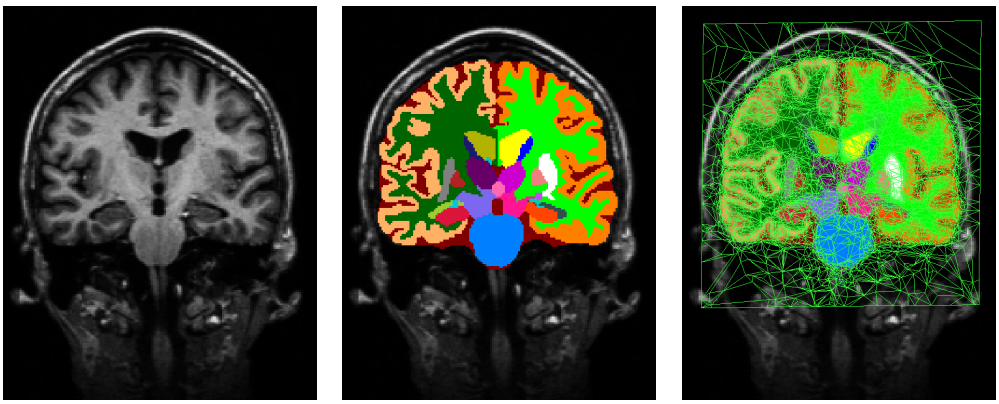


Figure 1: Left: T1-weighted scan from the training data. Center: corresponding manual segmentation. Right: atlas mesh built from 20 randomly selected subjects from the training data.

Table 1: Equations for the forward probabilistic model of MRI brain scans

| | | | |
|---|---|---|---|
| $\mathbf{x}$ | $\sim$ | $p(\mathbf{x})$ | (Eq. 2) |
| $\mathbf{l}$ | $\sim$ | $p(\mathbf{l}\|\mathbf{x})$ | (Eq. 3) |
| $\boldsymbol{\theta}$ | $\sim$ | $p(\boldsymbol{\theta}) \propto 1$ | |
| $\mathbf{D}$ | $\sim$ | $p(\mathbf{D}\|\mathbf{l}, \boldsymbol{\theta})$ | (Eq. 6) |

$$p_i(\mathbf{d}|k, \boldsymbol{\theta}) = \sum_{g=1}^{G_k} w_{k,g} \mathcal{N}\left(\mathbf{d} - \mathbf{C}\boldsymbol{\phi}^i | \boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}\right),$$

and

$$\mathcal{N}(\mathbf{d}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{d}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{d}-\boldsymbol{\mu})\right).$$

Here, $G_k$ is the number of Gaussian distributions in the mixture associated with label $k$; and $\boldsymbol{\mu}_{k,g}$, $\boldsymbol{\Sigma}_{k,g}$, and $w_{k,g}$ are the mean, covariance matrix, and weight of component $g \in \{1, \ldots, G_k\}$ in the mixture model of label $k$ (satisfying $w_{k,g} \geq 0$ and $\sum_g w_{k,g} = 1$). Furthermore,

$$\mathbf{C} = \begin{pmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_N^T \end{pmatrix}, \quad \mathbf{c}_n = \begin{pmatrix} c_{n,1} \\ \vdots \\ c_{n,P} \end{pmatrix} \quad \text{and} \quad \boldsymbol{\phi}^i = \begin{pmatrix} \phi_1^i \\ \phi_2^i \\ \vdots \\ \phi_P^i \end{pmatrix},$$

where $P$ denotes the number of bias field basis functions, $\phi_p^i$ is the basis function $p$ evaluated at voxel $i$, and $\mathbf{c}_n$ holds the bias field coefficients for MRI contrast $n$.

The entire forward model is summarized in Table 1.

### 2.3. Inference

Using the model described above, the MAP segmentation for a given MRI scan is obtained by maximizing Eq. 1 with respect to $\mathbf{l}$:

$$\hat{\mathbf{l}} = \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{D}) = \arg\max_{\mathbf{l}} p(\mathbf{D}|\mathbf{l}) p(\mathbf{l}), \tag{7}$$

which is intractable due to the integrals over the parameters $\mathbf{x}$ and $\boldsymbol{\theta}$ that appear in the expressions for $p(\mathbf{l})$ (Eq. 4) and $p(\mathbf{D}|\mathbf{l})$ (Eq. 5), respectively. This difficulty can be side-stepped if the posterior distribution of the model parameters in light of the data is heavily peaked around its mode:

$$p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{D}) \simeq \delta(\mathbf{x} - \hat{\mathbf{x}}, \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}),$$

where $\delta(\cdot)$ is Dirac's delta and the point estimates $\{\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}\}$ are given by:

$$\{\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}\} = \underset{\{\mathbf{x}, \boldsymbol{\theta}\}}{\operatorname{argmax}}\, p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{D}). \tag{8}$$

In that scenario, we can approximate:

$$\begin{aligned} p(\mathbf{l}|\mathbf{D}) &= \int_{\mathbf{x}} \int_{\boldsymbol{\theta}} p(\mathbf{l}|\mathbf{D}, \mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{D}) \mathrm{d}\mathbf{x} \mathrm{d}\boldsymbol{\theta} \\ &\simeq p(\mathbf{l}|\mathbf{D}, \hat{\mathbf{x}}, \hat{\boldsymbol{\theta}}), \end{aligned} \tag{9}$$

which no longer involves intractable integrals. The resulting inference algorithm then involves two distinct phases, detailed below: first, computing the point estimates by maximizing Eq. 8; and subsequently computing the segmentation by maximizing Eq. 9 with respect to $\mathbf{l}$.

*Computation of point estimates.* Applying Bayes' rule to Eq. 8, we obtain:

$$\begin{aligned} p(\mathbf{x},\boldsymbol{\theta}|\mathbf{D}) &\propto p(\mathbf{D}|\mathbf{x},\boldsymbol{\theta})p(\mathbf{x})p(\boldsymbol{\theta}) \\ &\propto \left(\sum_{\mathbf{l}} p(\mathbf{D}|\mathbf{l},\boldsymbol{\theta})p(\mathbf{l}|\mathbf{x})\right)p(\mathbf{x}) \\ &= \prod_{i=1}^{I}\left(\sum_{k=1}^{K} p_i(\mathbf{d}_i|k,\boldsymbol{\theta})p_i(k|\mathbf{x})\right)p(\mathbf{x}). \end{aligned}$$

Taking the logarithm, we can rewrite the problem as the maximization of the following objective function:

$$\{\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}\} = \underset{\{\mathbf{x},\boldsymbol{\theta}\}}{\operatorname{argmax}}\left[\sum_{i=1}^{I}\log\left(\sum_{k=1}^{K} p_i(\mathbf{d}_i|k,\boldsymbol{\theta})p_i(k|\mathbf{x})\right) + \log p(\mathbf{x})\right]. \tag{10}$$

We solve this problem with a coordinate ascent scheme, in which the mesh node positions $\mathbf{x}$ and likelihood parameters $\boldsymbol{\theta}$ are iteratively updated, by alternately optimizing one while keeping the other fixed.

To optimize the mesh node positions $\mathbf{x}$ with fixed $\boldsymbol{\theta}$, we use a standard conjugate gradient optimizer (Shewchuk, 1994). To optimize the likelihood parameters $\boldsymbol{\theta}$ with fixed $\mathbf{x}$, we use a generalized expectation-maximization (GEM) algorithm (Dempster et al., 1977) similar to the one proposed in (Van Leemput et al., 1999a). In particular, the GEM optimization involves iteratively computing the following soft assignments of each voxel to each of the Gaussian distributions, based on the current parameter estimates:

$$q_i^{k,g} = \frac{w_{k,g}\mathcal{N}\left(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i|\boldsymbol{\mu}_{k,g},\boldsymbol{\Sigma}_{k,g}\right)p_i(k|\mathbf{x})}{\sum_{k'=1}^{K} p_i(\mathbf{d}_i|k',\boldsymbol{\theta})p_i(k'|\mathbf{x})}, \tag{11}$$

and subsequently updating the parameters accordingly:

$$\boldsymbol{\mu}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i)}{\sum_{i=1}^{I} q_i^{k,g}}, \quad w_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}}{\sum_{i=1}^{I}\sum_{g'=1}^{G_k} q_i^{k,g'}},$$

$$\boldsymbol{\Sigma}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{k,g}(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)^T}{\sum_{i=1}^{I} q_i^{k,g}},$$

$$\begin{pmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_N \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{A}^T\mathbf{S}_{1,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{1,N} \\ \vdots & \ddots & \vdots \\ \mathbf{A}^T\mathbf{S}_{N,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{N,N} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mathbf{A}^T\left(\mathbf{S}_{1,1}\mathbf{r}_{1,1} + \dots + \mathbf{S}_{1,N}\mathbf{r}_{1,N}\right) \\ \vdots \\ \mathbf{A}^T\left(\mathbf{S}_{N,1}\mathbf{r}_{N,1} + \dots + \mathbf{S}_{N,N}\mathbf{r}_{N,N}\right) \end{pmatrix},$$

where

$$\mathbf{A} = \begin{pmatrix} \phi_1^1 & \dots & \phi_P^1 \\ \vdots & \ddots & \vdots \\ \phi_1^I & \dots & \phi_P^I \end{pmatrix}, \quad \mathbf{S}_{m,n} = \operatorname{diag}\left(s_i^{m,n}\right)$$

$$\text{and} \quad \mathbf{r}_{m,n} = \left(r_1^{m,n},\dots,r_I^{m,n}\right)^T, \quad \text{with}$$

$$s_i^{m,n} = \sum_{k=1}^{K}\sum_{g=1}^{G_k} s_{i,k,g}^{m,n}, \quad s_{i,k,g}^{m,n} = q_i^{k,g}\left(\boldsymbol{\Sigma}_{k,g}^{-1}\right)_{m,n}$$

$$r_i^{m,n} = d_i^n - \frac{\sum_{l=1}^{K}\sum_{g=1}^{G_l} s_{i,k,g}^{m,n}\left(\boldsymbol{\mu}_{k,g}\right)_n}{\sum_{l=1}^{K}\sum_{g=1}^{G_k} s_{i,k,g}^{m,n}}.$$

It can be shown that this process is guaranteed to increase the objective function of Eq. (10) with respect to $\boldsymbol{\theta}$ in each GEM iteration (Dempster et al., 1977; Van Leemput et al., 1999a).

*Computation of the final segmentation.* Given the point estimates of the model parameters, the conditional posterior distribution of the segmentation $\mathbf{l}$ factorizes over voxels:

$$p(\mathbf{l}|\mathbf{D},\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}) = \prod_{i=1}^{I} p_i(l_i|\mathbf{d}_i,\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}), \quad p_i(k|\mathbf{d}_i,\hat{\mathbf{x}},\hat{\boldsymbol{\theta}}) = \sum_{g=1}^{G_k} q_i^{k,g}.$$

The optimal segmentation for each voxel is therefore given by:

$$\hat{l}_i = \underset{k}{\operatorname{argmax}} \sum_{g=1}^{G_k} q_i^{k,g}.$$

### 2.4. Implementation

In practice, we have found that modeling substructures with similar intensity properties (e.g., all white matter structures) with the same Gaussian mixture model improves the robustness of the algorithm while giving faster execution times. Letting $s$ denote a set of structures that share the same mixture model, this is accomplished by altering the GEM update equations for the Gaussian mixture parameters as follows:

$$\boldsymbol{\mu}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{s,g}(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i)}{\sum_{i=1}^{I} q_i^{s,g}} \quad \forall k \in s,$$

$$w_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{s,g}}{\sum_{i=1}^{I}\sum_{g'=1}^{G_s} q_i^{s,g'}} \quad \forall k \in s,$$

$$\boldsymbol{\Sigma}_{k,g} \leftarrow \frac{\sum_{i=1}^{I} q_i^{s,g}(\mathbf{d_i} - \boldsymbol{\mu}_{s,g} - \mathbf{C}\boldsymbol{\phi}^i)(\mathbf{d_i} - \boldsymbol{\mu}_{s,g} - \mathbf{C}\boldsymbol{\phi}^i)^T}{\sum_{i=1}^{I} q_i^{s,g}} \forall k \in s,$$

where

$$q_i^{s,g} = \sum_{k\in s} q_i^{k,g}.$$

The details of which structures share the same mixture models will be given in Section 3.3.

To initialize the algorithm, we first affinely align the atlas to the target image using the registration method described in (D'Agostino et al., 2004), which uses atlas probabilities – rather than an intensity template – to drive the registration process. After the initial registration we mask out non-brain tissues by excluding voxels that have a prior probability lower than 0.01 of belonging to any of the brain structures.

The image intensities are then log-transformed to accommodate the additive bias field that is employed (cf. Section 2.2). For the bias field modeling, we use the lowest frequency components of the 3D discrete cosine transform (DCT) as basis functions (for the number of components see Section 3.3).

The subsequent optimization is done at two resolution levels. In the first level, the atlas probabilities are smoothed using a Gaussian kernel with a standard deviation of 2.0 mm in order to fit large scale mesh deformations. No smoothing is used in the second level, which refines the registration on a smaller scale.

The stopping criteria for the different components of the algorithm are as follows: the likelihood parameters $\boldsymbol{\theta}$ are updated until the relative change in the objective function (Eq. 10) falls under $10^{-5}$; the mesh node positions are updated until the maximum deformation across vertices falls under $10^{-3}$ mm; and the GEM and conjugate gradient optimizers are iteratively interleaved until the decrease in the cost function falls under $10^{-6}$.

The algorithm is implemented in Matlab except for the computationally demanding optimization of the mesh node positions, which is implemented in C++, and involves computing the mesh node deformation prior $p(\mathbf{x})$ (Eq. 2), the interpolated prior probabilities $p(\mathbf{l}|\mathbf{x})$ (Eq. 3) and the gradient of the objective function (Eq. 10) with respect to the mesh node positions.

## 3. Experiments

In this section, we first describe the brain MRI datasets used in this study (Section 3.1). Then, we outline four methods that our algorithm is benchmarked against (Section 3.2) . Next, we detail how the free parameters of each method are set (Section 3.3). Finally, we describe the setups for four different experiments in which the different methods are tested (Section 3.4).

### *3.1. MRI data*

In the experiments, we use five different sets of scans: one exclusively for training the segmentation methods, and the other four for testing the performance on unseen data. For training, we use a dataset of 39 T1-weighted MRI scans and corresponding expert segmentations obtained using a protocol described in (Caviness Jr et al., 1989). The data consists of 28 healthy subjects and 11 subjects with questionable or probable Alzheimer's disease with ages ranging from under 30 years old to over 60 years old (Sabuncu et al., 2010). The scans were acquired on a 1.5T Siemens Vision scanner using an MPRAGE sequence with parameters: TR=9.7ms, TE=4ms, TI=20ms, flip angle = 10° and voxel size = $1.0 \times 1.0 \times 1.5\text{mm}^3$ (128 sagittal slices), where the scan parameters were empirically optimized for gray-white matter contrast (Buckner et al., 2004). This is the same dataset used for training in the publicly available software package FreeSurfer (Fischl et al., 2002). An example scan and a corresponding manual segmentation are shown in Figure 1.

For testing, we use 219 scans from four different datasets acquired on scanners from different manufacturers, with different field strengths and pulse sequences. The first test dataset consists of 13 T1-weighted scans acquired on a 1.5T Siemens Sonata scanner with the same sequence and parameters as the training data (Han and Fischl, 2007). Given the similarity with the training data (vendor, field strength, pulse sequence), we will refer to this dataset as the **"intra-scanner dataset"**. The manual segmentations were obtained using the same protocol as for the training data. An example scan and a corresponding manual segmentation are shown in Figure 2.

The second test dataset consists of 14 T1-weighted scans acquired on a 1.5T GE Signa Scanner using an SPGR sequence with parameters: TR = 35 ms, TE = 5 ms, flip angle = 45° and voxel size = $0.9375 \times 0.9375 \times 1.5\text{mm}^3$ (124 coronal slices) (Han and Fischl, 2007). The manual segmentations were obtained using the same protocol as for the training data. This dataset will be referred to as the **"cross-scanner dataset"**. An example scan and a corresponding manual segmentation are shown in Figure 3.

The third test dataset consists of multi-echo FLASH scans from 8 healthy subjects acquired on a 1.5T Siemens Sonata scanner. The acquisition parameters were: TR = 20 ms, TE = min, flip angle = 3°, 5°, 20° and 30°, and voxel size = $1.0\text{mm}^3$ isotropic (Fischl et al., 2004b; Iglesias et al., 2012). The different flip angles correspond to different contrast properties, with the smallest angle having contrast similar to proton density (PD) weighting and the largest one having a contrast similar to T1-weighting. The manual segmentations were made using the same protocol as for

the training data. These data will be referred to as the **"multi-echo dataset"**. A sample slice from this dataset, with flip angles 30° and 3°, is shown in Figure 4.

The fourth and final test dataset consists of 40 healthy subjects scanned at two different time points at different facilities, with scan intervals ranging from 2 days to six months, amounting to a total of 80 T1- and T2-weighted scans for the whole dataset (Holmes et al., 2012). The scans were all acquired with 3T Siemens Tim Trio scanners using identical multi-echo MPRAGE sequences for the T1 and 3D T2-SPACE sequences for the T2, with voxel size = $1.2 \times 1.2 \times 1.2\text{mm}^3$. Note that the acquisition protocol was highly optimized for speed, with a total acquisition time for both scans of under 5 minutes. This dataset will be referred to as the **"test-retest dataset"**. One of the scans had to be excluded because of motion artifacts. Moreover, some of the T2-weighted scans have minor artifacts not present in the T1-weighted scans. These scans were however included in the experiments. Manual segmentations were not available for this dataset; however, these scans are still useful in test-retest experiments quantifying the differences between the two time points. Ideally, as all the subjects are healthy, the biological variations should be small and the segmentations between the two time points should be identical. An example of the T1- and T2-weighted scans is shown in Figure 5.

### *3.2. Benchmark methods*

In order to gauge the performance of the proposed algorithm with respect to the state-of-the-art in brain MRI segmentation, we compare its performance against four representative methods:

- **BrainFuse**[2] (Sabuncu et al., 2010) is a multi-atlas segmentation method, which uses an intensity-based label fusion approach to merge a set of propagated training labelings into a final segmentation of a target scan. More specifically, it assumes a generative model in which a latent, discrete membership field (whose smoothness is enforced by a Markov random field prior) indexes from which atlas the information was taken at each voxel. That information is corrupted with a probabilistic model (logOdds (Pohl et al., 2006a) for the labels, Gaussian noise for the intensities) to yield the test scan. Segmentation is carried out through Bayesian inference, using an iterative algorithm that alternatively: (1) uses local intensity and label pooling in

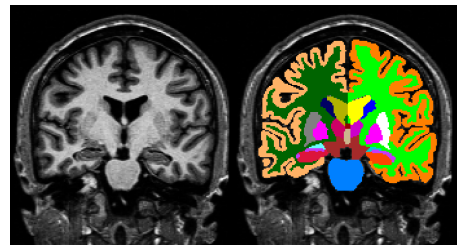


Figure 2: On the left an example slice from the intra-scanner dataset and on the right a corresponding manual segmentation.

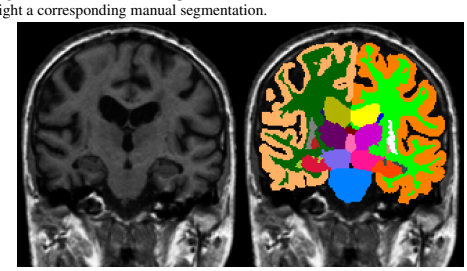


Figure 3: On the left an example slice from the cross-scanner dataset and on the right a corresponding manual segmentation.

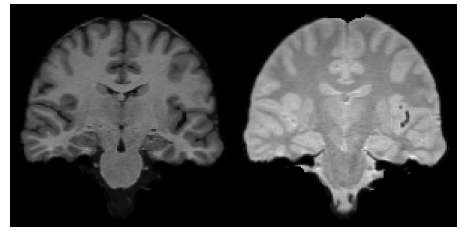


Figure 4: An example of the T1- (flip angle = 30°) and PD-weighted (flip angle = 3°) scans of the same subject from the multi-echo dataset.

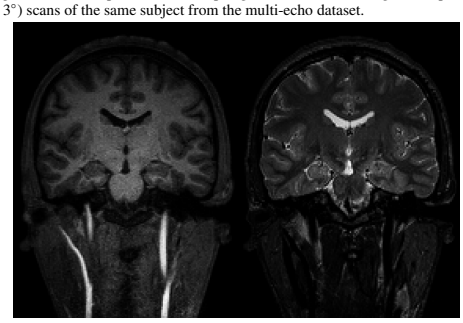


Figure 5: An example of the T1- and T2-weighted scans of the same subject from the test-retest dataset.

[2] `http://people.csail.mit.edu/msabuncu/sw/bfl/index.html`

the neighborhood of each voxel to compute a probabilistic estimate of the membership field; and (2) updates the segmentation by using the estimated field to weight the contribution of the atlases at each voxel. In the available implementation, the Markov random field smoothness prior is not included – however it does not yield a significant increase in segmentation accuracy (Sabuncu et al., 2010). For computing the registrations between the training and target subjects, BrainFuse employs asymmetric bidirectional registrations based on an efficient Demons-style algorithm that uses a one parameter sub-group of diffeomorphisms combined with a sum-of-squared-differences similarity measure (Sabuncu et al., 2010). The free parameters of the registration method are set to the values reported in (Sabuncu et al., 2010), where the authors cross-validated the parameter values on the same training dataset that we use in this study.

- **PICSL MALF**[3] (Wang et al., 2013) assumes that the segmentation errors of the propagated training labelings can be correlated, as opposed to BrainFuse, in which independence of the errors of the different labelings is assumed. PICSL MALF formulates a weighted voting problem in terms of trying to minimize the expectation of the labeling error, i.e., the error between the fused labels and the true segmentation in every voxel. To achieve this, it approximates the expected pairwise joint label differences between the training scans and the target scan using intensity similarity information. Moreover, PICSL MALF also performs a local search to try to find the voxel that is most similar to the corresponding target image voxel patch-wise. This can be interpreted as additional refinement of the pre-computed pairwise registrations. For computing the initial pair-wise registrations between the training and target subjects PICSL MALF uses ANTs/SyN[4] (Avants et al., 2008), which is a diffeomorphic registration algorithm. The registration parameters are set to the values which were used in the implementation of PICSL MALF that won the MICCAI 2012 Grand Challenge on Multi-Atlas Labeling (Landman and Warfield, 2012).

- **FreeSurfer**[5] (Fischl et al., 2002) is based on a statistical atlas of neuroanatomy, along with an intensity atlas in which a Gaussian distribution is associated with each voxel and class. The parameters of these Gaussians are estimated from training data. The model is completed by a Markov random field model which ensures the spatial smoothness of the segmentation, which is computed as the MAP estimate in a Bayesian framework. We note that FreeSurfer was trained on the same training data that we are using in this study, which makes direct comparison with our approach and the multi-atlas methods feasible.

- **Majority Voting** (Rohlfing et al., 2004; Heckemann et al., 2006) is a simple multi-atlas segmentation method, where the propagated training labelings are fused into a final segmentation by picking, in each voxel, the most frequent label across the propagated labelings. We include this method as a reference against which we can compare the performance of the more sophisticated label fusion approaches. For our implementation of majority voting, we use the same pair-wise registrations as for PICSL MALF.

These methods cover a wide spectrum of modern brain MRI segmentation algorithms. Majority voting, BrainFuse and PICSL MALF represent multi-atlas segmentation, which is arguably the most popular segmentation paradigm at the moment. Moreover, they are non-parametric methods, whereas our method and FreeSurfer represent parametric approaches. All four benchmark methods use supervision to model image intensities (i.e., intensity knowledge derived from the training scans is used to segment new scans), whereas the proposed method does not, allowing it to adapt to different MRI contrasts.

### *3.3. Cross-validation experiments on training data for parameter tuning*

The free parameters of the different methods are determined using the training dataset as follows:

*Proposed Algorithm.* We use 20 randomly picked subjects out of the available 39 to build our probabilistic atlas. Only 20 subject are chosen, because the atlas building process is very computationally expensive (several weeks to build an atlas with 20 subjects) and the results show that the segmentation performance does not increase any further when more subjects are added (see Section 4.3). The remaining 19 subjects are used to find suitable values for the free parameters in our algorithm: the global stiffness of the mesh $\beta$, the number

[3] `http://www.nitrc.org/projects/picsl_malf/`
[4] `http://stnava.github.io/ANTs/`
[5] `http://surfer.nmr.mgh.harvard.edu/`

Table 2: Details of the parameter sharing between structure classes. The groups of structures that share their Gaussian mixture parameters are shown in the first column, and the corresponding amount of Gaussians in the mixture in the second column.

| Structures with shared parameters | Number Of Gaussians |
|---|---|
| Non-brain tissues | 3 |
| L/R Cerebral White Matter (WM)<br>L/R Cerebellum White Matter (CWM)<br>Brain Stem (BS)<br>L/R Ventral Diencephalon<br>Optic Chiasm | 2 |
| L/R Cerebral Cortex (CT)<br>L/R Cerebellum Cortex (CCT)<br>L/R Caudate (CA)<br>L/R Hippocampus (HP)<br>L/R Amygdala (AM)<br>L/R Accumbens Area | 3 |
| L/R Lateral Ventricle (LV)<br>L/R Inferior Lateral Ventricle<br>3rd Ventricle<br>Cerebro-Spinal Fluid (CSF)<br>5th Ventricle<br>4th Ventricle<br>Vessel<br>L/R Choroid Plexus | 3 |
| L/R Thalamus (TH) | 2 |
| L/R Putamen (PU) | 2 |
| L/R Pallidum (PA) | 2 |

of bias field basis functions $P$, the groups of structures $s$ that share the same GMM parameters, and the number of mixture components associated with each structure group.

The parameters are tuned based on a visual inspection of the automatic segmentations. The chosen values for the mesh stiffness and number of bias field basis functions are: $\beta = 0.1$ and $P = 5$ per dimension, amounting to a total of $P = 5^3 = 125$ basis functions in 3D. The choice of which sets of structures share the Gaussian mixture parameters, as well as the number of Gaussians for each mixture, is summarized in Table 2.

*BrainFuse.* We use the optimal parameters listed in the original publication (Sabuncu et al., 2010); this choice is appropriate because the authors cross-validate the parameter values on the same training dataset as used in this study.

*PICSL MALF.* For this method we need to determine the optimal values for the patch radius over which the intensity similarity is calculated, a constant controlling the inverse distance function which maps the intensity difference to the joint error, and the size of the local search window (Wang et al., 2013). For this purpose, we randomly select 10 subjects as test data and use the remaining 29 subjects as training data, and perform a cross-validation grid search using similarity patch radii of $r_p = [1, 2, 3]$, local search radii of $r_s = [0, 1, 2, 3]$ and inverse mapping constants of $\beta = [0.5, 1, 1.5, 3, 6]$. As a measure of goodness we use the mean Dice overlap score[6] (which is the main performance metric used in the experiments below) over the structures listed in Section 3.4 below. The resulting optimal values are: $r_p = 1$, $r_s = 2$ and $\beta = 3$.

*FreeSurfer.* We use the standard processing pipeline with default parameters. No cross-validation needs to be performed as FreeSurfer is trained on the same training dataset we use in this study.

*Majority Voting.* Given the pre-computed registrations, majority voting has no parameters to tune.

### *3.4. Experimental setup*

We perform a comprehensive evaluation consisting of four sets of experiments:

I. In a first experiment, we use models trained on the training dataset to segment the scans from the intra-scanner and the cross-scanner datasets, comparing each method's segmentations with the corresponding manual annotations. This experiment enables us not only to compare the performance of the different methods, but also to assess how much their performance degrades when the image intensity properties of the training and test datasets are not matched.

II. In a second experiment, we evaluate the computational efficiency of the various methods. We compute the running time of the different algorithms on a cluster where each node has two quad-core Xeon 5472 3.0GHz CPUs and 32GB of RAM; we only use one core in the experiments in order to

[6] Dice $= 2|\mathbf{l}_A \cap \mathbf{l}_M|/(|\mathbf{l}_A| + |\mathbf{l}_M|)$, where $\mathbf{l}_A$ and $\mathbf{l}_M$ are the automatic and manual segmentations respectively and $|\cdot|$ is the cardinality of a set.

make fair comparisons, even though all the algorithms can potentially be parallelized. We also record the execution time of a multi-threaded implementation of our method, using 8 cores on a computer with 8 dual-cores with 3.4Ghz CPU and 64GB of RAM. This setup represents a realistic scenario that enables us to compare the running time of our algorithm with those reported by other studies in the literature.

III. In a third experiment, we study the effect of the number of training subjects on the segmentation performance. To achieve accurate segmentations, a representative training set is needed to capture all the structural variation one might see within the subjects to be segmented (Aljabar et al., 2009). However, some algorithms require less training data than others to approach their asymptotic performance, which represents a saving in manual labeling effort. We therefore randomly pick 5 sets of 5, 10 and 15 subjects from the training data, and re-evaluate the segmentation performance of the proposed method, BrainFuse, PICSL MALF and majority voting on the intra- and cross-scanner datasets.

IV. In a final experiment, we evaluate the ability of the proposed algorithm to segment multi-contrast MR scans in both the multi-echo and the test-retest dataset. Given a training set consisting only of T1-weighted scans, using multi-contrast information is out of reach for all the different methods we compare against in this article, either due to their non-parametric nature (Wang et al., 2013; Sabuncu et al., 2010; Heckemann et al., 2006) or intensity-dependent priors (Fischl et al., 2002). To quantify the effect of using multi-contrast information, we first run the proposed method using only one of the available scans and then using two scans with different contrasts. For the multi-echo dataset we first use only the T1-weighted images (i.e., flip angle 30°), and then both the T1- and PD-weighted (flip angle 3°) images. The automated segmentations are compared to the expert segmentations using Dice scores. For the test-retest dataset, in a similar fashion, we first segment the two time points using only the T1-weighted images and then using both T1- and T2-weighted images. Because no manual segmentations are available for this dataset, we use absolute symmetrized percent change (ASPC) (Reuter et al., 2012) to quantify the differences in the automatic segmentations between the two time points. This metric is defined as the absolute value of the difference in volume, normalized by the mean volume:

$$\mathrm{ASPC} = \frac{2|V_2 - V_1|}{V_1 + V_2},$$

where $V_1, V_2$ are the volumes at the two time points. Ideally this number should be small, as the subjects are all healthy and the time between the scans is not so long.

We report the Dice scores and the ASPC on a representative subset of 23 relevant structures which is also used in other studies (e.g., (Fischl et al., 2002; Sabuncu et al., 2010)): left and right cerebral white matter (WM), cerebellum white matter (CWM), cerebral cortex (CT), cerebellum cortex (CCT), lateral ventricle (LV), hippocampus (HP), thalamus (TH), putamen (PU), pallidum (PA), caudate (CA), amygdala (AM) and brain stem (BS). We will refer to these structures as the "regions of interest" (ROIs); note that for clarity of presentation we report the average Dice score of the left and right hemisphere for all structures except for the brain stem.

## 4. Results and discussion

### *4.1. Intra-scanner and cross-scanner segmentation performance*

The Dice scores between the manual and automated segmentations of the ROIs, obtained using the different methods, are shown for the intra-scanner dataset in Figure 6 (top). Table 3 (first column) summarizes the scores in average over the ROIs. All of the methods perform well on the intra-scanner dataset, which was expected, as the contrast-properties of the training data are identical to those of this dataset. The multi-atlas segmentation methods achieve the highest mean scores, with PICSL MALF being the best method for this dataset. Majority voting also obtains a very high mean score despite its simple fusion strategy. This is likely due to the accurate ANTs/SyN registration framework, which has been shown to perform very well on intra-scanner data (Klein et al., 2009). We note that each of the benchmark methods is specifically trained for this type of data, whereas the proposed method is not.

For the cross-scanner data, where the contrast-properties of the target data are different from the training data, the ROI Dice scores are shown in Figure 6 (bottom) and the mean scores over the ROIs in Table 3 (second column). The overall segmentation accuracy of all methods decreases, which is likely due to the lower intrinsic image contrast as a result of the different pulse

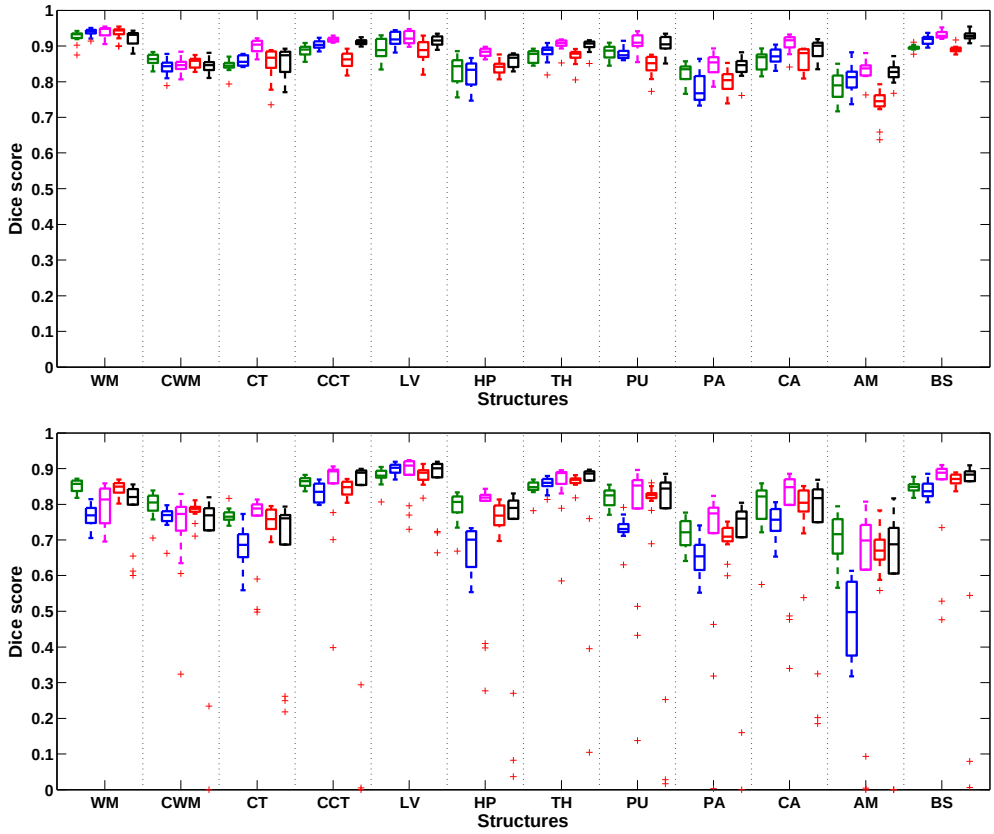



Figure 6: The Dice scores of the different methods for the intra-scanner (top) and cross-scanner (bottom) data. The proposed method = green, BrainFuse = blue, PICSL MALF = magenta, FreeSurfer = red and Majority Voting=black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'. See Section 3.4 for the acronyms.

| | Intra-scanner data | Cross-scanner data |
|---|---|---|
| Method | Average Accuracy | Average Accuracy |
| Proposed | 0.863 | **0.807** |
| BrainFuse | 0.868 | 0.744 |
| PICSL MALF | **0.896** | 0.760 |
| FreeSurfer | 0.853 | 0.799 |
| Majority Voting | 0.883 | 0.698 |

Table 3: Mean Dice scores of the different methods over the ROIs for the intra-scanner (first column) and cross-scanner (second column) datasets.

| | Average time per subject (h) | | |
|---|---|---|---|
| Method | Reg. | Fusion | Full Time |
| BrainFuse | 16 | 1 | 17 |
| Majority voting | 143.9 | 0.1 | 144 |
| PICSL MALF | 143.9 | 3.8 | 147.7 |
| FreeSurfer | - | - | 9.5 |
| Proposed | - | - | **1.4** |

Table 4: Mean computational time for the different methods. For label fusion methods the computation times for registration (Reg.) and label fusion (Fusion) are listed separately.

sequence as noted in (Han and Fischl, 2007). In this dataset, the proposed method achieves the highest mean score, demonstrating its robustness against changes in contrast. The label fusion methods, which rely on the image intensities of the training data in the registration and fusion steps, are clearly affected by the changes in the MRI contrast. The pair-wise registrations are especially more challenging for this dataset, due to the different intensity and resolution properties, which leads to misregistrations that are the principal error source in multi-atlas segmentation (Wang et al., 2013). Note that now majority voting performs the worst, as a result of the simple label fusion approach which can not downplay the effect of poorly registered subjects.

The relatively good performance of FreeSurfer, which also relies on the intensity information in the training scans, can be explained by its in-built renormalization procedure for T1 acquisitions, which applies a multi-linear atlas-image registration and a histogram matching step to update the class-conditional densities for each structure (Han and Fischl, 2007).

### *4.2. Running time*

The approximate mean computation time for a single scan using the different methods is shown in Table 4. The proposed method is approximately 7 times faster than FreeSurfer, 12 times faster than BrainFuse and 100 times faster than PICSL MALF and majority voting.

In general, the parametric methods (i.e., FreeSurfer and the proposed method) are significantly faster than the label fusion approaches. This is because only a single non-linear registration is needed, as opposed to the multiple pair-wise registrations used in the non-parametric methods. Moreover, in PICSL MALF the local search is especially time consuming with large search windows. Compared with FreeSurfer, which is also parametric, our method is faster due to the sparse encoding of the mesh prior. Encoding this sparsity is computationally expensive, but needs to be done only once (in an offline fashion). Furthermore, in the proposed approach, no special post or pre-processing of the target scans is needed.

| Number of subjects | Average number of vertices |
|---|---|
| 5 | 33,606 |
| 10 | 44,614 |
| 15 | 51,258 |

Table 5: Average number of vertices in the proposed atlas mesh for different numbers of training subjects.

In its multi-threaded setup, the proposed method has an execution time of 23.5 minutes per scan on average. The fastest whole-brain segmentation method to our knowledge is presented in (Zikic et al., 2014) with execution times in the range of 5 to 13 minutes; however this method is not designed to handle image contrast differences.

### *4.3. Effect of the number of training subjects*

Figure 7 shows the mean Dice scores over the ROIs, as well as their variance, across randomly selected sets of training subjects, plotted against the number of training subjects – for the intra-scanner and cross-scanner datasets. The results show that adding more training subjects generally yields more accurate segmentations for all methods, but that the proposed method reaches its maximum performance faster than the multi-atlas methods. Even with only five training subjects the segmentation accuracy of the proposed method is already good, with mean accuracy 98.5% of the maximal performance on the intra-scanner dataset and 96% of the maximal performance on the cross-scanner dataset. This is especially useful for populations where expert segmentations are expensive or difficult to obtain, such as infants. The variance of the mean score is also small on both datasets compared to the multi-atlas methods, indicating that the performance of the proposed method does not depend much on the specific subjects included in the training set. This is likely due to the atlas construction process that explicitly avoids over-fitting to training data (Van Leemput, 2009), yielding sparser tetrahedral meshes (and therefore blurrier probabilistic atlases) when less training subjects are available. This effect is illustrated in Table 5, where the average number of mesh vertices for the 5, 10 and 15 training subject groups are reported.

For the multi-atlas methods the performance is more dependent on the number of available training subjects, especially for the cross-scanner dataset. On the intra-scanner dataset, PICSL MALF achieves a good

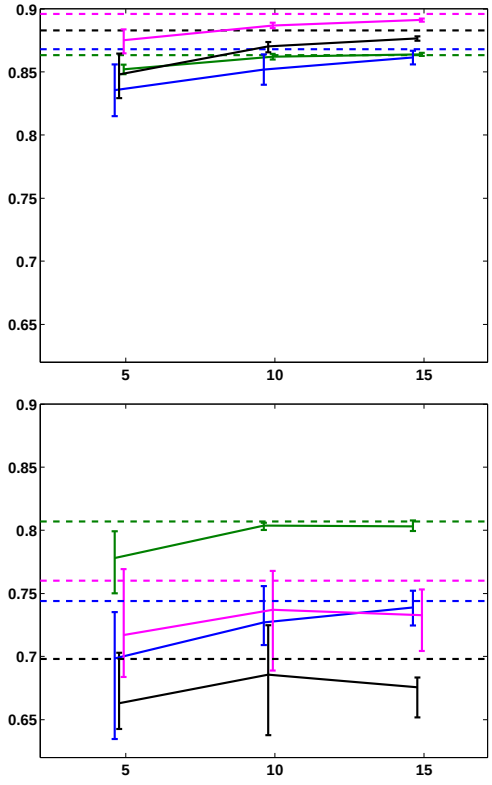



Figure 7: Average Dice scores for different number of training subjects for the intra-scanner (top) and the cross-scanner (bottom) data, as well as their variance across randomly selected sets of training subjects. The proposed method in green, BrainFuse in blue, PICSL MALF in magenta and majority voting in black. The error bars correspond to the lowest and highest average score for the random subset of subjects. The dashed line marks the Dice score obtained when all subjects in the training pool are used.

mean score already with 5 subjects, but the performance increases more slowly compared with the proposed method. The variance of the score is also larger, especially for the 5-subject set, showing that the performance is dependent on the particular subjects included in the training set. Majority voting and BrainFuse exhibit similar behaviour, but with larger variances over all the subjects sets. On the cross-scanner dataset the performance of all multi-atlas methods varies significantly even when trained on 15 subjects. This has been noted before in (Aljabar et al., 2009), where the authors suggest ways of pre-selecting a group of training subjects that are most similar to the target scan to increase performance of multi-atlas methods – particularly when majority voting is used and the contribution of poorly registered atlases cannot be downplayed by the label fusion algorithm.

### *4.4. Multi-contrast performance*

The Dice scores for the multi-echo dataset, when using only T1-weighted scans and when using both T1- and PD-weighted scans, are shown in Figure 8. The results are very similar whether or not the PD-weighted scan is included, indicating that the PD-weighted contrast does not add much useful information to the T1-weighted scan when healthy brains are segmented. Example segmentations of the multi-echo dataset using uni- and multi-contrast scans are shown in Figure 9.

The volume differences between the two time points in the 39 subjects of the T1/T2 test-retest dataset are shown in Figure 10. In general, they are quite similar and small for both single- (only T1) and multi-contrast (both T1 and T2) segmentations, with the median ASPC in the 1-2% range. There are some larger differences – especially in the thalamus and pallidum – when using multi-contrast data. This appears to be mostly due to imaging artifacts in the T2-scans, an example of which is shown in Figure 11. We note that this dataset has the lowest resolution of all the datasets we tested the method on, and therefore is affected the most by partial volume segmentation errors.

## 5. Conclusions

In this paper we have presented a whole-brain segmentation method that builds upon the parametric models commonly used in tissue classification. We have demonstrated that these type of models are capable of achieving state-of-the-art segmentation performance, while being very fast, adaptive to changes in tissue contrast, and able to handle multi-contrast data. We emphasize that the exact same algorithm was used for all datasets in this paper, without any parameter retuning or configuration changes, demonstrating the robustness of the approach.

Although in our experiments the method's segmentation accuracy trailed that of the very best multi-atlas techniques available today (PICSL MALF, cf. (Klein et al., 2009; Landman and Warfield, 2012)) in scenarios where the image intensities of the training and test datasets are perfectly matched, we found that the opposite is true when this is not the case. We believe the latter is a more realistic scenario in practice, since manual whole-brain segmentation is so time-consuming (e.g., taking hundreds of days for the training data used in this

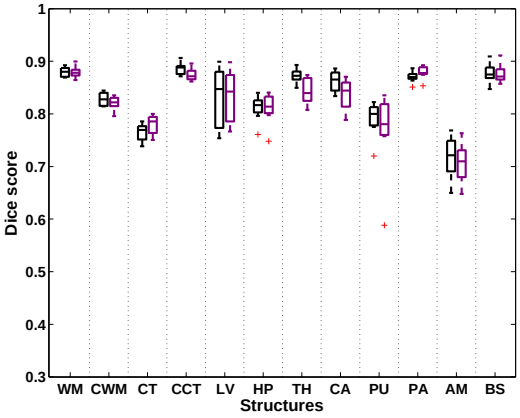



Figure 8: Dice scores for the multi-echo dataset. Performance on multi-contrast input data is shown in purple, and on T1-weighted data only in black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'.

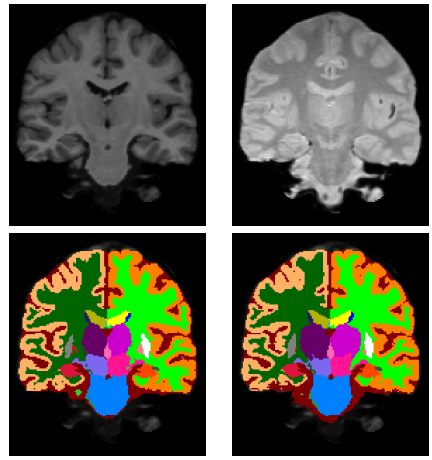


Figure 9: Top row: target scans, T1-weighted on the left and PD-weighted on the right. Bottom row: automatic segmentation using **only** the T1-weighted scan on the left, automatic segmentation using **both** scans on the right.

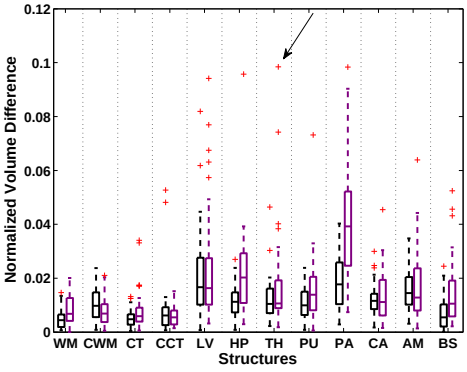



Figure 10: The ASPC scores for the test-retest dataset. Volume differences between the time points on multi-contrast input data is shown in purple, and on T1-weighted data only in black. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, and outliers are marked with a '+'. The outlier marked by an arrow is the one shown in Figure 11.

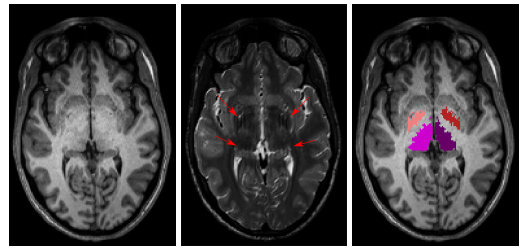


Figure 11: An example of an outlier subject marked by the arrow in Figure 10. From left to right: a T1-weighted scan with no visible artifacts, a T2-weighted scan with a line-like artifact in the pallidum and thalamus area marked by red arrows, and an automated segmentation of pallidum and thalamus showing the segmentation error caused by the artifact.

paper) that the available training data will seldom be acquired on the exact same imaging system as the images being segmented.

The proposed method has been evaluated on a set structures in which the cerebral cortex was considered a single structure, without attempting to further parcellate it into neuroanatomical subregions. However, we note that the volumetric white matter segmentations generated by the method can be used to build and label cortical surface models using FreeSurfer (Dale et al., 1999; Fischl et al., 2004a). Exploring this direction remains as future work.

In the current paper, we only analyzed images of healthy subjects, and our experiments on multi-contrast images showed no benefit in terms of segmentation accuracy compared to when only T1-weighted scans are used. However, the ability to seamlessly handle multi-contrast data becomes essential when analyzing diseased populations, since many brain lesions are much better visualized in T2-weighted and FLAIR scans than in T1-weighted contrast. In future work we will therefore include models of pathologies in the proposed framework, enabling simultaneous whole-brain segmentation and pathology detection.

## 6. Acknowledgements

This research was supported by the NIH NCRR (P41-RR14075, 1S10RR023043), NIBIB (R01EB013565), the Lundbeck foundation and financial contributions from the Technical University of Denmark. JEI acknowledges financial support from the Gipuzkoako Foru Aldundia (Fellows Gipuzkoa Program), as well as from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 654911.

## References

Aljabar, P., Heckemann, A.R., Hammers, A., Hajnal, V.J., Rueckert, D., 2009. Multi-atlas based segmentation of brain images: Atlas selection and its effect on accuracy. NeuroImage 46, 726–738.

Artaechevarria, X., Munõz Barrutia, A., Ortiz-de Solórzano, C., 2009. Combination strategies in multi-atlas image segmentation: Application to brain MR data. IEEE Transactions on Medical Imaging 28, 1266–1277.

Ashburner, J., Andersson, R.L.J., Friston, J.K., 2000. Image registration using a symmetric prior–in three dimensions. Human Brain Mapping 9, 212–225.

Ashburner, J., Friston, J.K., 1997. Multimodal image coregistration and partitioning – a unified framework. NeuroImage 6, 209–217.

Ashburner, J., Friston, J.K., 2005. Unified segmentation. NeuroImage 26, 839–885.

Asman, J.A., Landman, A.B., 2013. Non-local statistical label fusion for multi-atlas segmentation. Medical Image Analysis 17, 194–208.

Avants, B.B., Epstein, C.L., Grossman, M., Gee, J.C., 2008. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. Medical image analysis 12, 26–41.

Awate, S.P., Tasdizen, T., Whitaker, R.T., Foster, N., 2006. Adaptive Markov modeling for mutual-information-based, unsupervised MRI brain-tissue classification. Medical Image Analysis 10, 726–739.

Babalola, K.O., Patenaude, B., Aljabar, P., Schnabel, J., Kennedy, D., Crum, W., Smith, S., Cootes, T., Jenkinson, M., Rueckert, D., 2009. An evaluation of four automatic methods of segmenting the subcortical structures in the brain. Neuroimage 47, 1435–1447.

Bazin, P.L., Pham, D.L., 2008. Homeomorphic brain image segmentation with topological and statistical atlases. Medical Image Analysis 12, 616–625.

Buckner, R., Head, D., Parker, J., Fotenos, A., Marcus, D., Morris, J., Snyder, A., 2004. A unified approach for morphometric and functional data analysis in young, old, and demented adults using automated atlas-based head size normalization: Reliability and validation against manual measurement of total intracranial volumee. NeuroImage 23, 724–738.

Caviness Jr, V., Filipek, P., Kennedy, D., 1989. Magnetic resonance technology in human brain science: blueprint for a program based upon morphometry. Brain Dev. 11, 1–13.

Cootes, F.T., Edwards, J.G., Taylor, J.C., 1998. Active appearance models. Proceedings of the 5th European Conference on Computer Vision-Volume II , 484–498.

Coupé, P., Manjón, V.J., Fonov, V., Pruessner, J., Robles, M., Collins, L.D., 2011. Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. NeuroImage 54, 940–954.

D'Agostino, E., Maes, F., Vandermeulen, D., Suetens, P., 2004. Non-rigid atlas-to-image registration by minimization of class-conditional image entropy. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004 , 745–753.

D'Agostino, E., Maes, F., Vandermeulen, D., Suetens, P., 2006. A unified framework for atlas based brain image segmentation and registration, in: Biomedical Image Registration. volume 4057, pp. 136–143.

Dale, M.A., Fischl, B., Sereno, I.M., 1999. Cortical surface-based analysis I: Segmentation and surface reconstruction. NeuroImage 9, 179–194.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal Of The Royal Statistical Society, Series B 39, 1–38.

Fischl, B., van der Kouwe, A., Destrieux, C., Halgren, E., Ségonne, F., Salat, H.D., Busa, E., Seidman, J.L., Goldstein, J., Kennedy, D., Caviness, V., Makris, N., Rosen, B., Dale, M.A., 2004a. Automatically parcellating the human cerebral cortex. Cerebral Cortex 14, 11–22.

Fischl, B., Salat, D.H., van der Kouwe, A.J.W., Makris, N., Ségonne, F., Quinn, B.T., Dale, A.M., 2004b. Sequence-independent segmentation of magnetic resonance images. Neuroimage 23, S69–S84.

Fischl, B., Salat, H.D., Busa, E., Albert, M., Dieterich, M., Haselgrove, C., van der Kouwe, A., Killiany, R., Kennedy, D., Klaveness, S., Montillo, A., Makris, N., Rosen, B., Dale, M.A., 2002. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. Neuron 33, 341–355.

Han, X., Fischl, B., 2007. Atlas renormalization for improved brain MR image segmentation across scanner platforms. IEEE Transactions on Medical Imaging 26, 479–486.

Heckemann, R.A., Hajnal, J.V., Aljabar, P., Rueckert, D., Hammers, A., 2006. Automatic anatomical brain MRI segmentation combining label propagation and decision fusion. NeuroImage 33, 115–126.

Holmes, A.J., Lee, P.H., Hollinshead, M.O., Bakst, L., Roffman, J.L., Smoller, J.W., Buckner, R.L., 2012. Individual differences in amygdala-medial prefrontal anatomy link negative affect, impaired social functioning, and polygenic depression risk. The Journal of Neuroscience 32, 18087–18100.

Iglesias, J.E., Konukoglu, E., Zikic, D., Glocker, B., Van Leemput, K., Fischl, B., 2013a. Is synthesizing MRI contrast useful for inter-modality analysis?, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013, pp. 631–638.

Iglesias, J.E., Sabuncu, R.M., Van Leemput, K., 2012. A generative model for multi-atlas segmentation across modalities, in: 9th IEEE International Symposium on Biomedical Imaging (ISBI), 2012, pp. 888–891.

Iglesias, J.E., Sabuncu, R.M., Van Leemput, K., 2013b. A unified framework for cross-modality multi-atlas segmentation of brain MRI. Medical Image Analysis 17, 1181–1191.

Isgum, I., Staring, M., Rutten, A., Prokop, M., Viergever, M.A., van Ginneken, B., 2009. Multi-atlas-based segmentation with local decision fusion – application to cardiac and aortic segmentation in CT scans. Medical Imaging, IEEE Transactions on 28, 1000–1010.

Kelemen, A., Székely, G., Gerig, G., 1998. Three-dimensional model-based segmentation of brain MRI, in: Workshop on Biomedical Image Analysis, pp. 4–13.

Klein, A., Andersson, J., Ardekani, A.B., Ashburner, J., Avants, B., Chiang, M.C., Christensen, E.G., Collins, L.D., Gee, J., Hellier, P., Song, H.J., Jenkinson, M., Lepage, C., Rueckert, D., Thompson, P., Vercauteren, T., Woods, P.R., Mann, J.J., Parsey, V.R., 2009. Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. NeuroImage 46, 786–802.

Landman, A.B., Warfield, K.S., 2012. Miccai 2012 workshop on multi-atlas labeling, in: 15th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2012), pp. 91–95.

Ledig, C., Heckemann, A.R., Aljabar, P., Wolz, R., Hajnal, V.J., Hammers, A., Rueckert, D., 2012a. Segmentation of MRI brain scans using MALP-EM, in: MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labeling, pp. 79–82.

Ledig, C., Heckemann, R.A., Hammers, A., Lopez, J.C., Newcombe, V.F.J., Makropoulos, A., Lötjönen, J., Menon, D.K., Rueckert, D., 2015. Robust whole-brain segmentation: Application to traumatic brain injury. Medical Image Analysis 21, 40–58.

Ledig, C., Wolz, R., Aljabar, P., Lötjönen, J., Heckemann, R.A., Hammers, A., Rueckert, D., 2012b. Multi-class brain segmentation using atlas propagation and EM-based refinement, in: 9th IEEE International Symposium on Biomedical Imaging (ISBI), 2012, pp. 896–899.

Makropoulos, A., Gousias, I.S., Ledig, C., Aljabar, P., Serag, A., Hajnal, J.V., Edwards, A.D., Counsell, S.J., Rueckert, D., 2014. Automatic whole brain MRI segmentation of the developing neonatal brain. IEEE Transactions on Medical Imaging 33, 1818–1831.

Nyúl, L.G., Udupa, J.K., Zhang, X., 2000. New variants of a method of MRI scale standardization. Medical Imaging, IEEE Transactions on 19, 143–150.

Patenaude, B., Smith, M.S., Kennedy, N.D., Jenkinson, M., 2011. A Bayesian model of shape and appearance for subcortical brain segmentation. NeuroImage 56, 907–922.

Pizer, M.S., Fletcher, T.P., Sarang, J., Thall, A., Chen, Z.J., Fridman, Y., Fritsch, S.D., Gash, G.A., Glotzer, M.J., Jiroutek, R.M., Lu, C., Muller, E.K., Tracton, G., Yushkevich, P., Chaney, L.E., 2003. Deformable M-reps for 3D medical image segmentation. International Journal of Computer Vision 55, 85–106.

Pohl, K., Fisher, J., Shenton, M., McCarley, R., Grimson, W., Kikinis, R., Wells, W., 2006a. Logarithm odds maps for shape representation, in: Larsen, R., Nielsen, M., Sporring, J. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2006. Springer Berlin Heidelberg. volume 4191 of *Lecture Notes in Computer Science*, pp. 955–963.

Pohl, K.M., Bouix, S., Nakamura, M., Rohlfing, T., McCarley, R.W., Kikinis, R., Grimson, W. E., L., Shenton, M.E., Wells, W.M., 2007. A hierarchical algorithm for MR brain image parcellation. IEEE Transactions on Medical Imaging 26, 1201–1212.

Pohl, M.K., Fisher, J., Grimson, L.E.W., Kikinis, R., Wells, M.W., 2006b. A Bayesian model for joint segmentation and registration. NeuroImage 31, 228–239.

Prastawa, M., Gerig, G., Lin, W., Gilmore, J.H., 2005. Automatic segmentation of MR images of the developing newborn brain. Medical Image Analysis 9, 457–466.

Puonti, O., Iglesias, J.E., Van Leemput, K., 2013. Fast, sequence adaptive parcellation of brain MR using parametric models, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013, pp. 727–734.

Reuter, M., Schmansky, J.N., Rosas, D.J., Fischl, B., 2012. Within-subject template estimation for unbiased longitudinal image analysis. NeuroImage 61, 1402–1418.

Rohfling, T., Brandt, R., Menzel, R., Maurer Jr., R.C., 2004a. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. NeuroImage 21, 1428–1442.

Rohfling, T., Russakoff, B.D., Maurer Jr., R.C., 2004b. Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation. IEEE Transactions on Medical Imaging 23, 983–994.

Rohlfing, T., Brandt, R., Menzel, R., Maurer, C.R., 2004. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. NeuroImage 21, 1428–1442.

Rousseau, F., Habas, A.P., Studholme, C., 2011. A supervised patch-based approach for human brain labeling. IEEE Transaction on Medical Imaging 30, 1852–1862.

Roy, S., Carass, A., Prince, J., 2013. Magnetic resonance image example-based contrast synthesis. IEEE Transactions on Medical Imaging 32, 2348–2363.

Sabuncu, M.R., Yeo, T.T.B., Van Leemput, K., Fischl, B., Golland, P., 2010. A generative model for image segmentation based on label fusion. IEEE Transactions on Medical Imaging 29, 1714–1729.

Shewchuk, J.R., 1994. An introduction to the conjugate gradient method without the agonizing pain. Technical Report.

Ta, V.T., Giraud, R., Collins, D.L., Coupé, P., 2014. Optimized patch-match for near real time and accurate label fusion, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014, pp. 105–112.

Tong, T., Wolz, R., Coupé, P., Hajnal, V.J., Rueckert, D., 2013. Segmentation of MR images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. NeuroImage 76, 11–23.

Van Leemput, K., 2009. Encoding probabilistic brain atlases using Bayesian inference. IEEE Transactions on Medical Imaging 28, 822–837.

Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P., 1999a. Automated model-based bias field correction of MR images of the brain. IEEE Transactions on Medical Imaging 18, 897–908.

Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P., 1999b. Automated model-based tissue classification of MR images of the brain. IEEE Transactions on Medical Imaging 18, 885–896.

Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P., 2001. Au-

tomated segmentation of multiple sclerosis lesions by model outlier detection. IEEE Transactions on Medical Imaging 20, 677–688.

Wang, H., Suh, W.J., Das, R.S., Pluta, J., Craige, C., Yushkevich, A.P., 2013. Multi-atlas segmentation with joint label fusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 35, 611–623.

Wells, M.W., III, Grimson, L.E.W., Kikinis, R., Jolesz, A.F., 1996. Adaptive segmentation of MRI data. IEEE Transactions on Medical Imaging 15, 429–442.

Wu, G., Wang, Q., Zhang, D., Nie, F., Huang, H., Shen, D., 2013. A generative probability model of joint label fusion for multi-atlas based brain segmentation. Medical Image Analysis .

Zhang, Y., Brady, M., Smith, S., 2001. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. Medical Imaging, IEEE Transactions on 20, 45–57.

Zijdenbos, A.P., Forghani, R., Evanc, A.C., 2002. Automatic "pipeline" analysis of 3-D MRI data for clinical trials: Application to multiple sclerosis. IEEE Transactions on Medical Imaging 21, 1280–1291.

Zikic, D., Glocker, B., Criminisi, A., 2014. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Medical Image Analysis 18, 1262–1273.

Chapter 9

# Paper C

# Simultaneous Whole-Brain Segmentation and White Matter Lesion Detection Using Contrast-Adaptive Probabilistic Models

Oula Puonti[1] and Koen Van Leemput[1,2]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark
[2] Martinos Center for Biomedical Imaging, MGH, Harvard Medical School, USA

**Abstract.** In this paper we propose a new generative model for simultaneous brain parcellation and white matter lesion segmentation from multi-contrast magnetic resonance images. The method combines an existing whole-brain segmentation technique with a novel spatial lesion model based on a convolutional restricted Boltzmann machine. Unlike current state-of-the-art lesion detection techniques based on discriminative modeling, the proposed method is not tuned to one specific scanner or imaging protocol, and simultaneously segments dozens of neuroanatomical structures. Experiments on a public benchmark dataset in multiple sclerosis indicate that the method's lesion segmentation accuracy compares well to that of the current state-of-the-art in the field, while additionally providing robust whole-brain segmentations.

## 1 Introduction

Conditions that affect the integrity of the white matter, including small vessel disease and multiple sclerosis, form a significant health concern. Lesions in the white matter are frequently associated with memory impairment, headaches, depression, muscle weakness, and many other conditions. Because magnetic resonance (MR) imaging can visualize lesion formation with much greater sensitivity than clinical observation, the ability to reliably and efficiently detect white matter lesions from MR scans is of great value to diagnose disease, track progression, and evaluate treatment. Quantifying the independent contribution of white matter lesions to clinical disability is important for enhancing our understanding of disease mechanisms, and for facilitating efficient testing in clinical trials.

Because of considerable intra- and inter-rater variabilities in manual annotations, and because of the sheer amount of imaging data acquired in clinical trials, there is a strong need for computational tools that can analyze brain images with white matter lesions in a fully automated fashion. Although many partial solutions have been proposed (e.g., [1]), a generally applicable tool that works robustly across disease states and imaging centers remains an open problem. Many of the best performing methods for lesion segmentation currently use

extended spatial neighborhoods to provide rich contextual information, using a *discriminative* approach in which the specific intensity characteristics of training images are explicitly used to encode the relationship between image appearance and segmentation labels (e.g., [2–4]). However, because of the dependency of MR intensity contrast on the scanner platform and pulse sequence, and because there exists no standardized clinical MR protocol to study white matter damage, such discriminative methods do not generalize well to cases where the target and training data come from different scanners or centers. Furthermore, these methods do not provide segmentations of the non-lesioned parts of the brain into various cortical and subcortical structures, although regional atrophy patterns convey vital clinical information in diseases such as multiple sclerosis [5].

In this paper, we propose a novel method for jointly segmenting white matter lesions and a large number of cortical and subcortical structures from multi-contrast MR data. The method combines a previously validated method for whole-brain segmentation of healthy brain scans [6] with a novel spatial model for lesion shape and occurrence that is conditioned on surrounding neuroanatomy. In particular we propose to use a restricted Boltzmann machine (RBM) [7] to provide much richer spatial models than the low-order Markov random fields (MRFs) that have traditionally been used in the field for spatial regularization of lesion segmentations [8]. By using a generative rather than a discriminative formulation, the method is able to completely separate models of anatomy (which are learned from manual segmentations of training data) from intensity models (which are estimated on the fly for each individual scan being segmented). Because the *intensities* of training data are never used, the model can be applied to images with new contrast properties without needing new training data.

We test our approach on publicly available data from the MICCAI 2008 MS lesion segmentation challenge [9], demonstrating the feasibility of the method. Compared to related work for simultaneous whole-brain and lesion segmentation [10], the proposed method segments considerably more structures, and learns spatial lesion models automatically from training data rather than relying on a set of hand-crafted rules to remove false positive detections.

## 2 Modeling Framework

We build upon a previously published generative modeling approach [6], in which a forward probabilistic image model is "inverted" to obtain automated segmentations. In the following we first briefly summarize the existing whole-brain segmentation method we build upon; then introduce the proposed RBM lesion model; describe how we integrate it within the model for whole-brain segmentation; and specify how we use the resulting model to obtain automated segmentations.

### 2.1 Existing whole-brain segmentation method

Let $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_I)$ denote a matrix collecting the (log-transformed) intensities in a multi-contrast brain MR scan with $I$ voxels, where the vector $\mathbf{d}_i =$

$(d_i^1, \ldots, d_i^N)^T$ contains the intensities in voxel $i$ for each of the available $N$ contrasts. Furthermore, let $\mathbf{l} = (l_1, \ldots, l_I)^T$ be the corresponding segmentation, where $l_i \in \{1, \ldots, K\}$ denotes the one of $K$ possible segmentation labels assigned to voxel $i$. A generative model then consists of a prior segmentation probability $p(\mathbf{l})$ that encodes prior knowledge about human neuroanatomy, and a segmentation-conditional probability $p(\mathbf{D}|\mathbf{l})$ that measures how probable the observed MR intensities are for different segmentations. In [6] the segmentation prior is parametrized by a sparse tetrahedral mesh with node positions $\boldsymbol{\theta}_l$. Assuming conditional independence of the labels between voxels given $\boldsymbol{\theta}_l$, the prior is given by:

$$p(\mathbf{l}) = \int_{\boldsymbol{\theta}_l} p(\mathbf{l}|\boldsymbol{\theta}_l) p(\boldsymbol{\theta}_l) \mathrm{d}\boldsymbol{\theta}_l, \quad \text{where}$$
$$p(\mathbf{l}|\boldsymbol{\theta}_l) = \prod_{i=1}^{I} p_i(l_i|\boldsymbol{\theta}_l)$$

and $p(\boldsymbol{\theta}_l)$ is a topology-preserving deformation prior. The prior model is learned from manual annotations in 39 subjects as described in [6].

For the segmentation-conditional distribution $p(\mathbf{D}|\mathbf{l})$, a Gaussian mixture model (GMM) is associated with each neuroanatomical label to model the relationship between segmentation labels and image intensities. The smoothly varying intensity inhomogeneities ("bias fields") that typically corrupt MR scans are modeled as a linear combination of spatially smooth basis functions that are added to the local voxel intensities. Letting $\boldsymbol{\theta}_d$ denote all bias field and GMM parameters with prior $p(\boldsymbol{\theta}_d) \propto 1$, the resulting segmentation-conditional distribution is given by:

$$p(\mathbf{D}|\mathbf{l}) = \int_{\boldsymbol{\theta}_d} p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}_d) p(\boldsymbol{\theta}_d) \mathrm{d}\boldsymbol{\theta}_d, \quad \text{where}$$
$$p(\mathbf{D}|\mathbf{l}, \boldsymbol{\theta}_d) = \prod_{i=1}^{I} p_i(\mathbf{d}_i|l_i, \boldsymbol{\theta}_d) \quad \text{and}$$
$$p_i(\mathbf{d}|l, \boldsymbol{\theta}_d) = \sum_{g=1}^{G_l} w_{lg} \mathcal{N}\big(\mathbf{d} - \mathbf{C}^T \boldsymbol{\phi}^i \big| \boldsymbol{\mu}_{lg}, \boldsymbol{\Sigma}_{lg}\big).$$

Here $\mathcal{N}(\cdot)$ denotes a normal distribution; $G_l$ is the number of Gaussian distributions associated with label $l$; and $\boldsymbol{\mu}_{lg}$, $\boldsymbol{\Sigma}_{lg}$, and $w_{lg}$ are the mean, covariance, and weight of component $g$ in the corresponding mixture model. Furthermore, $\boldsymbol{\phi}^i$ evaluates the bias field basis functions at the $i^{\text{th}}$ voxel, and $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_N)$ where $\mathbf{c}_n$ denotes the parameters of the bias field model for the $n^{\text{th}}$ MR contrast.

With this model segmentation proceeds by estimating $\hat{\mathbf{l}} = \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{D})$, using the approximation $p(\mathbf{l}|\mathbf{D}) \simeq p(\mathbf{l}|\mathbf{D}, \hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l)$ where $\{\hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l\}$ are the parameter values that maximize $p(\boldsymbol{\theta}_d, \boldsymbol{\theta}_l|\mathbf{D})$. These values are estimated using coordinate ascent, where the atlas deformation parameters $\boldsymbol{\theta}_l$ are optimized with a conjugate gradient (CG) algorithm, and the remaining parameters $\boldsymbol{\theta}_d$ with a

generalized expectation-maximization (GEM) algorithm [6]. The optimization is done iteratively in an alternating fashion keeping the deformation parameters fixed while optimizing the intensity model parameters and vice versa until convergence. The GMM parameters are initialized based on the structure probabilities given by the segmentation prior model after affine registration to the target scan. We emphasize that the intensity model parameters are learned *given* the target scan and thus automatically adapt to its intensity properties. In [6] the intensity-adaptiveness was demonstrated on several datasets acquired with different sequences, scanners and field strengths.

### 2.2 Spatial lesion prior using a convolutional RBM

In order to model the spatial configuration of white matter lesions, we employ a restricted Boltzmann machine (RBM) [7], a specific type of MRF in which long-range voxel interactions are encoded through local connections to hidden units, which effectively function as feature detectors. Letting $\mathbf{z} = (z_1, \ldots, z_I)^T$ denote a binary lesion map, where $z_i \in \{0, 1\}$ indicates if the voxel is part of a lesion, a RBM prior on $\mathbf{z}$ is defined by

$$p(\mathbf{z}) = \sum_{\mathbf{h}} p(\mathbf{z}, \mathbf{h}), \quad \text{with}$$
$$p(\mathbf{z}, \mathbf{h}) \propto \exp\big[ - E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h})\big],$$

where $\mathbf{h} = (h_1, \ldots, h_J)^T, h_j \in \{0, 1\}$ denotes a vector of $J$ binary hidden units, and the RBM "energy" is defined as:

$$E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h}) = -\mathbf{b}^T\mathbf{z} - \mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{z}.$$

The parameters of this model include the vectors $\mathbf{b}$ and $\mathbf{c}$ (which bias individual visible and hidden units to take on certain values), as well as the weight matrix $\mathbf{W}$ (which models the interaction between the hidden and visible units). The attractiveness of this specific MRF model arises from the presence of the hidden units, which increase the expressive power of the model, as well as the property that the values of $\mathbf{z}$ are independent of one another given $\mathbf{h}$ and vice versa, which greatly facilitates inference computations. Specifically, for each hidden unit $h_j$ and lesion $z_i$ the conditional distributions are written as [11]:

$$p(h_j = 1|\mathbf{z}) = \sigma\left(c_j + \big(\mathbf{W}\mathbf{z}\big)_j\right)$$
$$p(z_i = 1|\mathbf{h}) = \sigma\left(b_i + \big(\mathbf{h}^T\mathbf{W}\big)_i\right),$$

where $\sigma(x) = (1 + exp(-x))^{-1}$.

In order to scale this framework to model full-sized images, we use a convolutional approach that imposes a repeated, sparse spatial structure on the parameters [11]. For the sake of clarity of presentation, in the following we describe the case for one-dimensional images, although the technique generalizes readily into three dimensions. In the convolutional RBM a set of $P$ filters

$\{\mathbf{f}^p\}_{p=1}^P, \mathbf{f}^p = (f_1^p, \ldots, f_Q^p)^T$ is defined, each of size $Q \ll I$. The parameter matrix $\mathbf{W}$ is then restricted to be of the form

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}^1 \\ \vdots \\ \mathbf{W}^P \end{pmatrix}, \quad \text{where} \quad \mathbf{W}^p = \begin{pmatrix} f_1^p & \ldots & f_Q^p & 0 & \ldots & 0 \\ 0 & f_1^p & \ldots & f_Q^p & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & f_1^p & \ldots & f_Q^p \end{pmatrix},$$

so that each filter detects the same specific feature in different parts of the image, and inference can be done efficiently using convolution. Similarly, in the parameter vector $\mathbf{c}$ each filter output shares the same bias across the image [11]. In our implementation we do not put such a restriction on the visible biases $\mathbf{b}$, as this allows modeling spatially varying prior probabilities of lesion occurrence.

We automatically learn appropriate values for the parameters $\{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$ from manually annotated training data, i.e., binary lesion maps for a number of different subjects. For this purpose, we use the persistent contrastive divergence (PCD) learning algorithm, which performs stochastic gradient ascent on the log-likelihood of the training data using approximate gradients computed with Markov chain Monte Carlo (MCMC) sampling [12].

### 2.3 Joint model

We incorporate the RBM lesion model into the whole-brain segmentation framework by assuming that a lesion can only occur in a voxel when its underlying neuroanatomical label is white matter ($l = \text{wm}$), effectively changing its status from healthy white matter ($z = 0$) into white matter lesion ($z = 1$). Towards this end, we define a joint segmentation prior on both $\mathbf{l}$ and $\mathbf{z}$:

$$p(\mathbf{l}, \mathbf{z}) = \int_{\boldsymbol{\theta}_l} p(\mathbf{l}, \mathbf{z}|\boldsymbol{\theta}_l) p(\boldsymbol{\theta}_l) \mathrm{d}\boldsymbol{\theta}_l, \quad \text{where}$$

$$p(\mathbf{l}, \mathbf{z}|\boldsymbol{\theta}_l) = \sum_{\mathbf{h}} p(\mathbf{l}, \mathbf{z}, \mathbf{h}|\boldsymbol{\theta}_l) \quad \text{and}$$

$$p(\mathbf{l}, \mathbf{z}, \mathbf{h}|\boldsymbol{\theta}_l) \propto \exp\left[ - E_{\mathrm{RBM}}(\mathbf{z}, \mathbf{h}) + \sum_{i=1}^{I} \log p_i(l_i|\boldsymbol{\theta}_l) - \sum_{i=1}^{I} \phi(l_i, z_i) \right],$$

where in abuse of notation $p_i(l_i|\boldsymbol{\theta}_l)$ refers to the deformable atlas of the whole-brain segmentation model, and $\phi(l, z)$ evaluates to zero when $l = \text{wm}$ or $z = 0$, and infinity otherwise. The role of $\phi(l, z)$ is to restrict lesions to appear only inside white matter – without it the model would devolve into simply $p(\mathbf{l}, \mathbf{z}) = p(\mathbf{l})p(\mathbf{z})$. In similar vein, we define an intensity model which is conditional on

both $\mathbf{l}$ and $\mathbf{z}$:

$$p(\mathbf{D}|\mathbf{l},\mathbf{z}) = \int_{\boldsymbol{\theta}_d} p(\mathbf{D}|\mathbf{l},\mathbf{z},\boldsymbol{\theta}_d)p(\boldsymbol{\theta}_d)\mathrm{d}\boldsymbol{\theta}_d, \quad \text{where}$$

$$p(\mathbf{D}|\mathbf{l},\mathbf{z},\boldsymbol{\theta}_d) = \prod_{i=1}^{I} p(\mathbf{d}_i|l_i,z_i,\boldsymbol{\theta}_d) \quad \text{and}$$

$$p_i(\mathbf{d}|l,z,\boldsymbol{\theta}_d) = \sum_{g=1}^{G_l} w_{lg}\mathcal{N}\left(\mathbf{d} - \mathbf{C}^T\boldsymbol{\phi}^i|\boldsymbol{\mu}_{lg},\gamma^z\boldsymbol{\Sigma}_{lg}\right).$$

This model preserves the original segmentation-conditional GMMs for voxels without lesions ($z = 0$), but widens the variances of the Gaussian components by a user-specified factor $\gamma > 1$ otherwise. Such wide distributions aim to capture the fact that lesions often do not have a clearly defined intensity profile in MR, e.g., ranging from iso-intense to white matter to intensities similar to CSF in T1-weighted contrasts.

### 2.4 Inference

Segmentation with the proposed model can be accomplished by first estimating the parameters $\{\hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l\}$ that maximize $p(\boldsymbol{\theta}_d, \boldsymbol{\theta}_l|\mathbf{D})$, and subsequently analyzing $p(\mathbf{l},\mathbf{z}|\mathbf{D},\hat{\boldsymbol{\theta}}_d,\hat{\boldsymbol{\theta}}_l)$, as in the original segmentation method. However, optimization of the model parameters is now complicated by the fact that the RBM model introduces non-local dependencies between the voxels through the weighted connections between the lesions and the hidden units. To side-step this difficulty, during the parameter estimation phase – in which we have no interest in accurately segmenting the white matter lesions – we temporarily replace the RBM energy $E_{\mathrm{RBM}}(\mathbf{z},\mathbf{h})$ with a simple energy of the form:

$$E_{\mathrm{tmp}}(\mathbf{z},\mathbf{l}) = -\sum_{i=1}^{I}[l_i = \mathrm{wm}]\big(z_i \log(1-w) + (1-z_i)\log w\big),$$

where $0 \leq w \leq 1$ is a user-specified parameter which essentially defines a uniform spatial prior probability for lesions to occur *within* white matter. This effectively removes the hidden units from the model, and reduces the form of $p(\boldsymbol{\theta}_d, \boldsymbol{\theta}_l|\mathbf{D})$ to the one of the original segmentation method, so that the same optimization strategy can be used. Compared to the original method, the only difference is that each Gaussian distribution $\mathcal{N}\big(\cdot\,|\boldsymbol{\mu}_{lg},\boldsymbol{\Sigma}_{lg}\big)$ associated with the white matter label $l = \mathrm{wm}$ is replaced with a mixture of the form:

$$(1-w)\mathcal{N}\big(\cdot\,|\boldsymbol{\mu}_{lg},\boldsymbol{\Sigma}_{lg}\big) + w\mathcal{N}\big(\cdot\,|\boldsymbol{\mu}_{lg},\gamma\boldsymbol{\Sigma}_{lg}\big), \tag{1}$$

yielding a distribution with the same mean but heavier tails, making parameter estimation more robust to intensity outliers such as white matter lesions. The adaptation in the GEM algorithm to enforce the parameter sharing between the two mixture components in Eq.(1) is straightforward.

Once the optimal parameter estimates are found, we replace the temporary energy with the original RBM energy and infer the corresponding whole-brain and lesion segmentation by MCMC sampling from $p(\mathbf{l}, \mathbf{z}|\mathbf{D}, \hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l)$, exploiting the specific structure of the RBM model. In particular, we generate $S$ triplets $\{\mathbf{l}_s, \mathbf{z}_s, \mathbf{h}_s\}_{s=1}^S$ by sampling from the distribution $p(\mathbf{l}, \mathbf{z}, \mathbf{h}|\mathbf{D}, \hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l)$ using block-Gibbs sampling. This is straightforward to implement because each of the conditional distributions factorizes over the voxels (for $\mathbf{l}$ and $\mathbf{z}$) or the hidden units (for $\mathbf{h}$). The sampling is performed in two alternating steps: first, we sample the values for the hidden units given the lesions:

$$\mathbf{h}_s \sim \prod_{j=1}^{J} p(h_j = 1|\mathbf{z}_{s-1}).$$

Next, given the sampled hidden unit values $\mathbf{h}_s$, we jointly sample the labels $\mathbf{l}$ and $\mathbf{z}$ from:

$$\{\mathbf{l}_s, \mathbf{z}_s\} \sim p(\mathbf{l}, \mathbf{z}|\mathbf{D}, \mathbf{h}_s, \hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l).$$

This is a multinomial distribution with $K + 1$ labels, where the label in each voxel is sampled from:

$$p(l_i, z_i|\mathbf{D}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{x}}, \mathbf{h}) \propto \begin{cases} p(\mathbf{d}_i|l_i, z_i = 0, \hat{\boldsymbol{\theta}}) p(l_i|\hat{\mathbf{x}}) p(z_i = 0|\mathbf{h}) \\ p(\mathbf{d}_i|l_i = \text{wm}, z_i = 1, \hat{\boldsymbol{\theta}}) p(l_i = \text{wm}|\hat{\mathbf{x}}) p(z_i = 1|\mathbf{h}) \\ 0, \quad l_i \neq \text{wm} \quad \text{and} \quad z_i = 1 \end{cases}$$

and $p(z_i = 0|\mathbf{h}) = 1 - p(z_i = 1|\mathbf{h})$. The initial lesion segmentation, i.e., $\mathbf{z}_0$, is obtained as a maximum-a-posteriori estimate using the temporary energy $E_{\text{tmp}}$.

Once we have acquired $S$ triplets, the samples of the hidden units $\{\mathbf{h}_s\}$ are discarded as they are of no interest to us. The "hard" segmentations of $\mathbf{l}$ and $\mathbf{z}$ are obtained by voxel-wise majority voting across $\{\mathbf{l}_s\}$ and $\{\mathbf{z}_s\}$.

## 3 Experiments and Results

### 3.1 Data

We demonstrate the proposed method on the 20 publicly available training cases of the MICCAI 2008 challenge on multiple sclerosis lesion segmentation [9]. This dataset includes 10 subjects scanned at Children's Hospital Boston (CHB) and another 10 scanned at the University of North Carolina (UNC). For each subject the scan set consists of a T1-weighted, a T2-weighted and a FLAIR scan with isotropic resolution of 0.5mm, along with expert segmentations provided by CHB[3]. As a pre-processing step the data was downsampled by a factor of two to a resolution of 1mm isotropic as is customary for this dataset [13, 2, 4]. No further pre- or post-processing, such as intensity normalization or bias field correction, was applied.

[3] Manual segmentations from UNC are now also available, but at the time of the challenge this was not the case [9] so we decided to use only the segmentations provided by CHB.

### 3.2 Implementation

We closely follow the implementation details of the whole-brain segmentation method described in [6]. Because of the small number of manual segmentations available for training the RBM model, we applied two rotations of 10 and $-10$ degrees around the three main axes, producing 6 extra training scans per subject. We trained different RBM models with either $P = 20$ or $P = 40$ filters, with sizes of $(Q \times Q \times Q)$, where $Q$ was either 5, 7 or 9. Each model was trained with 5600 gradient steps of size 0.1 in the PCD algorithm [12]. Based on pilot experiments, we found that using two mixture components for white matter worked well (i.e., $G_{\text{wm}} = 2$), provided that one of the Gaussians is constrained to be a near-uniform distribution that can collect model outliers other than white matter lesions (in practice we use a Gaussian with a fixed scalar covariance matrix $10^6\mathbf{I}$ and weight 0.05). Finally, as the main characteristic of white matter lesions is that they appear hyper-intense compared to normal white matter in FLAIR contrast [14], we decided to only allow voxels to be assigned to lesion in the Gibbs sampling process if their intensity is higher than the estimated white matter mean in FLAIR.

We implemented the algorithm in Matlab, except for the mesh deformation part, which was written in C++, and the RBM convolutions, which were performed on a GPU. In our experiments, estimation of the parameters $\{\hat{\boldsymbol{\theta}}_d, \hat{\boldsymbol{\theta}}_l\}$ was performed on a cluster where each node has two quad-core Xeon 5472 3.0GHz CPUs and 32GB of RAM. Only one core was used in the experiments, taking roughly 1.7 hours per subject. Gibbs sampling was done on a machine with a GeForce GTX Titan 6GB GPU. We generated $S = 150$ samples, collected after an initial burn-in of 50 sampling steps, taking approximately 10 minutes per subject. Thus the full segmentation time for a single target scan is roughly two hours.

### 3.3 Evaluation set-up

In order to compare our results against previous methods on the same data, we use the true positive rate $\text{TPR} = \frac{\text{TP}}{\text{TP+FN}}$ and the positive predictive value $\text{PPV} = \frac{\text{TP}}{\text{TP+FP}}$ as performance metrics. Here TP, FP and FN count the true positive, false positive and false negative voxels compared to the expert segmentation. Because our method contains four user-specified parameters $\gamma$, $w$, $Q$ and $P$, which can have a large influence on the obtained results, and because the RBM requires training data to learn its parameters, we perform our evaluation in a cross-validation setting. In particular, we split the available data randomly into five distinct sets, each having 16 training and 4 test subjects. For segmenting each set of 4 test subjects, the remaining 16 are used to train the RBM and to find the best combination $(\gamma, w, Q, P)$, defined as the combination maximizing the product of the mean TPR and PPV over the 16 subjects. Using the product as a measure of fitness promotes parameter combinations which provide both sensitive and specific lesion segmentations.

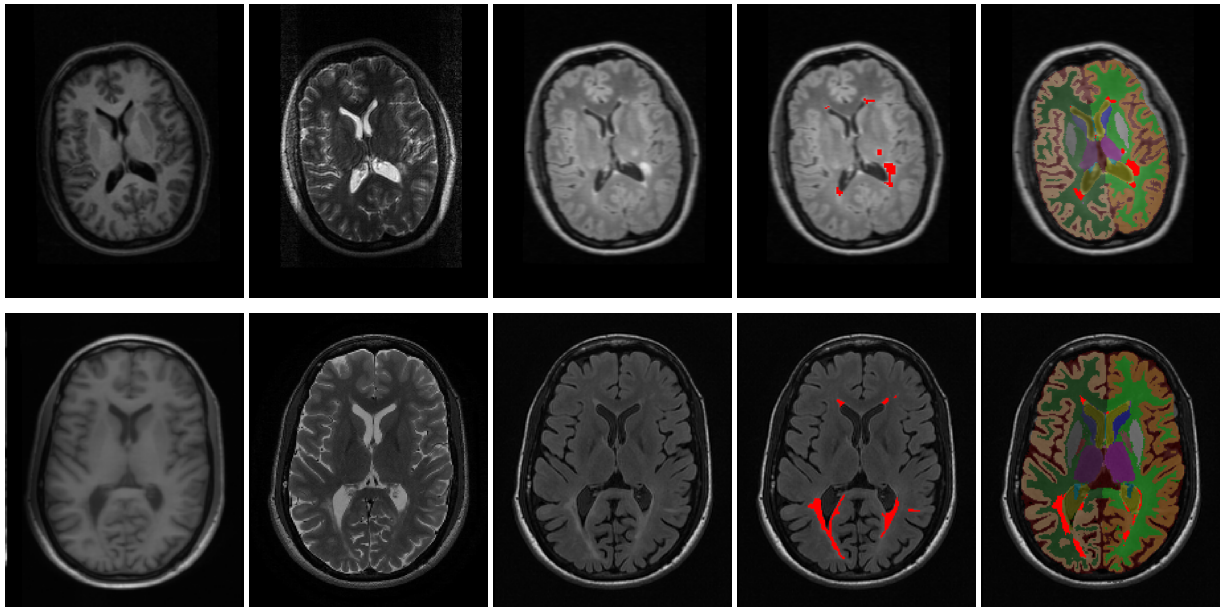


**Fig. 1.** Example segmentations from two subjects CHB04 (first row) and CHB08 (second row). From left to right: T1-weighted scan, T2-weighted scan, FLAIR scan, manual segmentation overlaid on the FLAIR scan, and the full segmentation obtained using the proposed method. Lesions are denoted in red.

### 3.4 Results

Figure 1 shows two examples of the joint whole-brain and lesion segmentations obtained using the proposed method, along with the manual segmentations. Although our method can segment 41 different neuroanatomical structures in total [6], the MICCAI challenge data only includes manual segmentations of lesions, so validation of the automatic segmentations of these structures could not be performed. However, visual inspection of the 20 cases did not reveal any significant failures in the whole-brain segmentation component of the method.

In table 1 we compare our lesion segmentation performance with that of two state-of-the-art lesion segmentation tools: a random forest (RF) classifier [2], which is a discriminative model, and a dictionary-learning approach (DL) [13], which is unsupervised and therefore contrast-adaptive (as is the proposed method). Compared to the winning method [15] of the MICCAI 2008 lesion segmentation challenge, which obtained a mean TPR of 0.21 and a mean PPV of 0.30, all the methods show greatly improved segmentation results. On average the proposed method achieves better results than both the DL and RF approaches, although the improvement over the RF approach is very slight. We note that neither of the two benchmark methods segments other structures than lesions, and that the RF classifier is specifically trained on the contrast properties of this specific data set, and is therefore less generally applicable than the proposed and DL methods. Note that the results of the DL method are not entirely comparable, as the authors used a different set of manual annotations for validating the UNC subjects. This explains the quite large difference in performance of the DL method compared to the two others for subjects UNC01 and UNC06.

In a very recently published work [4], the authors present a lesion segmentation framework based on deep convolutional encoder networks. This model is somewhat similar to the proposed method in the sense that both use convolutional architectures for learning suitable features for lesion detection automatically. The authors also report results on the MICCAI 2008 dataset, obtaining an average TPR of 0.40 and an average PPV of 0.41 which ties the performance of the proposed method. However, their approach is, similar to the RF, based on a discriminative classifier and only segments lesions.

## 4 Discussion

In this paper we have proposed a method for joint white matter lesion detection and whole-brain segmentation using a novel spatial lesion model. Due to the generative modeling approach, the method is not tied to one specific scanner platform or imaging protocol, and shows good performance when compared to the current state-of-the-art in lesion segmentation. The presented results are significantly limited by the amount of training data, which was very small given the number of parameters and potential expressive power of the RBM model. Future work will involve further experimentation with different RBM training algorithms and sampling strategies, and an extensive performance validation on

| | DL [13] | | RF [2] | | Proposed | | | DL [13] | | RF [2] | | Proposed | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Patient | TPR | PPV | TPR | PPV | TPR | PPV | Patient | TPR | PPV | TPR | PPV | TPR | PPV |
| CHB01 | 0.60 | 0.58 | 0.49 | **0.64** | **0.75** | 0.57 | UNC01 | **0.33** | **0.29** | 0.02 | 0.01 | 0.02 | 0.01 |
| CHB02 | 0.27 | 0.45 | 0.44 | **0.63** | **0.57** | 0.48 | UNC02 | 0.54 | **0.51** | 0.48 | 0.36 | **0.75** | 0.29 |
| CHB03 | 0.24 | 0.56 | 0.22 | 0.57 | **0.30** | **0.69** | UNC03 | **0.64** | **0.27** | 0.24 | 0.35 | 0.28 | 0.19 |
| CHB04 | 0.27 | 0.66 | 0.31 | **0.78** | **0.59** | 0.49 | UNC04 | 0.40 | **0.51** | 0.54 | 0.38 | **0.62** | 0.40 |
| CHB05 | 0.29 | 0.33 | 0.40 | **0.52** | **0.45** | 0.39 | UNC05 | 0.25 | 0.10 | **0.56** | **0.19** | 0.50 | 0.18 |
| CHB06 | 0.10 | 0.36 | **0.32** | **0.52** | 0.19 | 0.50 | UNC06 | 0.13 | **0.55** | 0.15 | 0.08 | **0.17** | 0.10 |
| CHB07 | 0.14 | 0.48 | **0.40** | 0.54 | 0.34 | **0.65** | UNC07 | 0.44 | 0.23 | **0.76** | 0.16 | 0.60 | **0.26** |
| CHB08 | 0.21 | **0.73** | **0.46** | 0.65 | 0.37 | 0.70 | UNC08 | 0.43 | 0.13 | **0.52** | **0.32** | 0.27 | 0.21 |
| CHB09 | 0.05 | 0.22 | **0.23** | 0.28 | 0.04 | **0.55** | UNC09 | **0.69** | 0.06 | 0.67 | **0.36** | 0.67 | 0.21 |
| CHB10 | 0.15 | 0.12 | **0.23** | 0.39 | 0.19 | **0.69** | UNC10 | 0.43 | 0.23 | **0.53** | 0.34 | 0.47 | **0.48** |

| | DL [13] | RF [2] | Proposed |
|---|---|---|---|
| Mean | TPR=0.33 PPV=0.37 | TPR=0.40 PPV=**0.40** | TPR=**0.41** PPV=**0.40** |

**Table 1.** Quantitative comparison with two state-of-the-art methods.

larger data sets of white matter lesions. We further plan to also validate the healthy structure segmentations obtained using the model.

**Acknowledgements:** This research was supported by NIH NCRR (P41-RR14075), NIBIB (R01EB013565), the Lundbeck Foundation (R141-2013-13117), and financial contributions from the Technical University of Denmark.

## References

1. Tomas-Fernandez, X., Warfield, S.: A model of population and subject (MOPS) intensities with application to multiple sclerosis lesion segmentation. Medical Imaging, IEEE Transactions on **34**(6) (June 2015) 1349–1361
2. Geremia, E., Clatz, O., Menze, B.H., Konukoglu, E., Criminisi, A., Ayache, N.: Spatial decision forests for MS lesion segmentation in multi-channel magnetic resonance images. NeuroImage **57**(2) (2011) 378–390
3. Karimaghaloo, Z., Rivaz, H., Arnold, D., Collins, D., Arbel, T.: Adaptive voxel, texture and temporal conditional random fields for detection of gad-enhancing multiple sclerosis lesions in brain MRI. In: Medical Image Computing and Computer-Assisted Intervention MICCAI 2013. Volume 8151 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2013) 543–550
4. Brosch, T., Yoo, Y., Tang, L., Li, D., Traboulsee, A., Tam, R.: Deep convolutional encoder networks for multiple sclerosis lesion segmentation. In: Medical Image Computing and Computer-Assisted Intervention MICCAI 2015. Volume 9351 of Lecture Notes in Computer Science. Springer International Publishing (2015) 3–11
5. Filippi, M., Rocca, M.A., Arnold, D.L., Bakshi, R., Barkhof, F., De Stefano, N., Fazekas, F., Frohman, E., Wolinsky, J.S.: EFNS guidelines on the use of neuroimaging in the management of multiple sclerosis. European Journal of Neurology **13**(4) (2006) 313–325
6. Puonti, O., Iglesias, J., Van Leemput, K. Lecture Notes in Computer Science. In: Fast, Sequence Adaptive Parcellation of Brain MR Using Parametric Models. Springer (2013) 727–734

7. Smolensky, P.: Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. MIT Press (1986) 194–281
8. Van Leemput, K., Maes, F., Vandermeulen, D., Colchester, A., Suetens, P.: Automated segmentation of multiple sclerosis lesions by model outlier detection. Medical Imaging, IEEE Transactions on **20**(8) (2001) 677–688
9. Styner, M., Lee, J., Chin, B., Chin, M., Commowick, O., Tran, H., Markovic-Plese, S., Jewells, V., Warfield, S.: 3D segmentation in the clinic: A grand challenge II: MS lesion segmentation. The MIDAS Journal (11 2008) 1–5
10. Shiee, N., Bazin, P.L., Ozturk, A., Reich, D.S., Calabresi, P.A., Pham, D.L.: A topology-preserving approach to the segmentation of brain images with multiple sclerosis lesions. NeuroImage **49**(2) (2010) 1524–1535
11. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning, ACM (2009) 609–616
12. Tieleman, T.: Training restricted Boltzmann machines using approximations to the likelihood gradient. In: Proceedings of the 25th international conference on Machine learning. (2008) 1064–1071
13. Weiss, N., Rueckert, D., Rao, A.: Multiple sclerosis lesion segmentation using dictionary learning and sparse coding. In: Medical Image Computing and Computer-Assisted Intervention MICCAI 2013. Volume 8149 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2013) 735–742
14. García-Lorenzo, D., Francis, S., Narayanan, S., Arnold, D.L., Collins, D.L.: Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. Medical image analysis **17**(1) (2013) 1–18
15. Souplet, J., Lebrun, C., Ayache, N., Malandain, G.: An automatic segmentation of T2-FLAIR multiple sclerosis lesions. http://hdl.handle.net/10380/1451 (07 2008)

Chapter 10

# Paper D

# Brain Tumor Segmentation by a Generative Model with a Prior on Tumor Shape

Mikael Agn[1], Oula Puonti[1], Ian Law[2], Per Munck af Rosenschöld[3] and Koen Van Leemput[1,4]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark
[2] Department of Clinical Physiology, Nuclear Medicine and PET, and
[3] Department of Oncology, Rigshospitalet, Denmark
[4] Martinos Center for Biomedical Imaging, MGH, Harvard Medical School, USA

**Abstract.** We present a fully automated generative method for brain tumor segmentation in multi-modal magnetic resonance images. We base the method on the type of generative model often used for healthy brain tissues, where tissues are modeled by Gaussian mixture models combined with a spatial tissue prior. We extend the basic model with a tumor prior, which uses convolutional restricted Boltzmann machines to model tumor shape. Experiments on the 2015 and 2013 BRATS data sets indicate that the method's performance is comparable to the current state of the art in the field, while being readily extendable to any number of input contrasts and not tied to any specific imaging protocol.

## 1 Introduction

Brain tumor segmentation from magnetic resonance (MR) images is of high value in radiosurgery and radiotherapy planning. Automatic tumor segmentation is challenging since tumor location, shape and appearance vary greatly across patients. Moreover, brain tumor images often exhibit significant intensity inhomogeneity as well as large intensity variations between subjects, particularly when they are acquired with different scanners or at different imaging facilities.

Most current state-of-the-art methods exploit the specific intensity contrast information of annotated training images, which hinders their applicability to images acquired with different imaging protocols. In this paper we propose an automated generative method that achieves segmentation accuracy comparable to the state of the art while being contrast-adaptive and readily extendable to any number of input contrasts. To achieve this, we incorporate a prior on tumor shape into an atlas-based probabilistic model for healthy tissue segmentation. The prior models tumor shape by convolutional restricted Boltzmann machines (RBMs) that are trained on expert segmentations, without the use of the *intensity information* corresponding to these segmentations.

## 2 Generative modeling framework

Let $\mathbf{D} = (\mathbf{d}_1, ..., \mathbf{d}_I)$ denote the multi-contrast MR data, where $I$ is the number of voxels and $\mathbf{d}_i$ contains the intensities at voxel $i$. We aim to segment each voxel

$i$ into either a healthy tissue label $l_i \in \{1, ..., K\}$ or tumor tissue $z_i \in \{0, 1\}$ and within tumor tissue into either edema or core $y_i \in \{0, 1\}$. For this purpose we build a generative model that describes the image formation and then use this model to derive a fully automated segmentation algorithm. To avoid cluttered equations we define the model in 1D; it is easily extended to the 3D images we actually use. We use the posterior of all variables given the data:

$$p(\mathbf{l}, \mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}, \boldsymbol{\theta}|\mathbf{D}) \propto p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta}) \cdot p(\mathbf{l}) \cdot p(\boldsymbol{\theta}) \cdot p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}). \quad (1)$$

The model consists of a likelihood function $p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta})$, which links labels to MR intensities, and priors $p(\mathbf{l})$, $p(\boldsymbol{\theta})$ and $p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G})$, where $\mathbf{H}$ and $\mathbf{G}$ denotes the hidden units of the RBMs (see further below). We define the likelihood as

$$p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta}) = \prod_i \begin{cases} p(\mathbf{d}_i|l_i, \boldsymbol{\theta}_l) & \text{if } z_i = 0 \text{ and } y_i = 0, \text{ (healthy tissue)} \\ p(\mathbf{d}_i|\boldsymbol{\theta}_e) & \text{if } z_i = 1 \text{ and } y_i = 0, \text{ (edema)} \\ p(\mathbf{d}_i|\boldsymbol{\theta}_c) & \text{if } z_i = 1 \text{ and } y_i = 1, \text{ (core)} \end{cases}, \quad (2)$$

where $\boldsymbol{\theta}$ contains the unknown model parameters $\boldsymbol{\theta}_l$, $\boldsymbol{\theta}_e$, $\boldsymbol{\theta}_c$ and bias field parameters $\mathbf{C}$ and $\boldsymbol{\phi}$; and $p(\mathbf{d}_i|l, \boldsymbol{\theta}_l) = \sum_{lg} \gamma_{lg} \mathcal{N}(\mathbf{d}_i - \mathbf{C}^T \boldsymbol{\phi}^i | \boldsymbol{\mu}_{lg}, \boldsymbol{\Sigma}_{lg})$ is a Gaussian mixture model (GMM). Subscript $g$ denotes a Gaussian component within label $l$ and $\mathcal{N}(\cdot)$ denotes a normal distribution; and $\gamma_{lg}$, $\boldsymbol{\mu}_{lg}$ and $\boldsymbol{\Sigma}_{lg}$ are the weight, mean and covariance of the corresponding Gaussian. The probabilities $p(\mathbf{d}_i|\boldsymbol{\theta}_e)$ and $p(\mathbf{d}_i|\boldsymbol{\theta}_c)$ are also GMMs. Furthermore, bias fields corrupting the MR scans are modeled as linear combinations of spatially smooth basis function added to the scans [4]. $\boldsymbol{\phi}^i$ contains basis functions at voxel $i$ and $\mathbf{C} = (\mathbf{c}_1, ..., \mathbf{c}_n)$, where $\mathbf{c}_n$ denotes the parameters of the bias field model for MR contrast $n$.

We use a probabilistic affine atlas computed from segmented healthy subjects as the healthy tissue prior [5], defined as $p(\mathbf{l}) = \prod_i \pi_{li}$. The atlas includes probability maps of GM, WM, CSF and background (BG). Moreover, we add a prior $p(\boldsymbol{\theta})$ on the distribution parameters [6], which ensures that the Gaussians modeling tumor tissue are neither too narrow or too wide and that their mean values in FLAIR are higher than that of $\boldsymbol{\mu}_{GM}$.

**Tumor prior:** We model tumor shape by convolutional RBMs, which are graphical models over visible and hidden units that allow for efficient sampling over large images without a predefined size [1]. The energy term of an RBM is defined as $E(\mathbf{z}, \mathbf{H}) = -\sum_k \mathbf{h}_k \bullet (\mathbf{w}_k * \mathbf{z}) - \sum_k b_k \sum_j h_j^k - c \sum_i z_i$, where $\bullet$ denotes element-wise product followed by summation and $*$ denotes convolution. Each hidden group $\mathbf{h}_k$ is connected to the visible units in $\mathbf{z}$ with a convolutional filter $\mathbf{w}_k$. To lower the amount of parameters to be estimated, we let each element in $\mathbf{w}_k$ model two neighboring elements in $\mathbf{z}$, e.g. a filter of size 7 will span over 14 voxels in $\mathbf{z}$. Furthermore, each hidden group has a bias $b_k$ and $\mathbf{z}$ a bias $c$.

We separately train one RBM for the complete tumor label $\mathbf{z}$ and one RBM for the tumor core label $\mathbf{y}$, where we estimate the filters and bias terms from training data. This is done by stochastic gradient ascent with contrastive divergence approximation of the log-likelihood gradients with one Gibbs sample step

[2]. We use the enhanced gradient to obtain more distinct filters [3]. After the training phase we combine the two RBMs to form the tumor shape prior:

$$p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}) \propto \mathrm{e}^{-E(\mathbf{z},\mathbf{H})-E(\mathbf{y},\mathbf{G})-f(\mathbf{y},\mathbf{z})}. \tag{3}$$

For each voxel, $f(y_i, z_i) = \infty$ if $y_i = 1$ and $z_i = 0$, and otherwise 0. This restricts tumor core tissue to only exist within the complete tumor.

**Inference:** We initially estimate $\boldsymbol{\theta}$ by a generalized Expectation-Maximization algorithm (GEM), where the tumor shape prior's energy is replaced with a simple energy of the form: $-\sum_i [l_i \neq BG](z_i \log w + (1-z_i)\log(1-w))$. This reduces the model to the same as in [4] with the addition of $p(\boldsymbol{\theta})$. We set $w$ to the expected fraction of tumor tissue within brain tissue, estimated from training data. After the initial parameter estimation, we fix the bias field parameters and infer the remaining variables by block-Gibbs Markov chain Monte Carlo sampling (MCMC). This is straightforward to implement as each of the conditional distributions $p(\mathbf{l}, \mathbf{z}, \mathbf{y}|\mathbf{D}, \mathbf{H}, \mathbf{G}, \boldsymbol{\theta})$, $p(\mathbf{H}|\mathbf{z})$, $p(\mathbf{G}|\mathbf{y})$ and $p(\boldsymbol{\theta}|\mathbf{D}, \mathbf{l}, \mathbf{z}, \mathbf{y})$ factorizes over its components. The MCMC is initialized with a *maximum a posteriori* (MAP) segmentation after GEM. After a burn-in period, we collect samples of $\mathbf{l}, \mathbf{z}$ and $\mathbf{y}$ and perform a voxel-wise majority voting across the collected samples.

## 3 Experiments

We used the training data of the BRATS 2013 challenge (30 subjects) as our training data set and tested the proposed method on the two test sets of 2013 (Leaderboard: 25 subjects, Challenge: 10 subjects) [7] and the training data of the 2015 BRATS challenge (274 subjects, some are re-scans). The data include four MR-sequences: FLAIR, T1, T2 and contrast-enhanced T1, and ground truth segmentations. All data have previously been skull-stripped.

**Implementation:** We used 40 filters of size $(7 \times 7 \times 7)$ for each RBM, trained with 9600 gradient steps of size 0.1, which took around 3 days each. To extend the training data, the tumor segmentations were flipped in 8 directions.

We registered the healthy tissue atlas by an affine transformation and log-transformed the MR intensities, to account for the additive bias field model [4]. We represented the core label $\mathbf{y}$ with one Gaussian during GEM, corresponding to enhanced core, and two during MCMC, one for enhanced core and one for remaining core. Before MCMC, the remaining core Gaussian was initialized by randomly setting $y_i = 1$ to a fraction of the voxels with $z_i = 1$ and $y_i = 0$ in the MAP segmentation. The fraction was chosen so that the total fraction of core within the complete tumor equaled the average fraction in the training data set. All other labels were represented by one Gaussian each, except CSF and BG that were represented with two Gaussians each.

Due to the large size variation of tumors, we found it beneficial to alter the bias term $c$ connected to $\mathbf{z}$ to better represent the tumor to be segmented. Before MCMC, we added $\log\left(\frac{p_{zs}(1-p_{zt})}{p_{zt}(1-p_{zs})}\right)$ to $\mathbf{c}$, where $p_{zs}$ denotes the fraction of tumor

within the GEM-segmented brain and $p_{zt}$ denotes the average tumor size in the training data set, used to train the RBM. We altered the bias term connected to $\mathbf{y}$ in the same way, with the difference that we instead used the average fraction of core within complete tumor in the training data set.

The full segmentation algorithm took approximately 30 minutes per subject. We generated 15 samples after a burn-in of 200. All computations were done on a i7-5930K CPU and a GeForce GTX Titan Black GPU in MATLAB 2014b.

**Results:** At the time of writing, our method is ranked in the top-5 of all submitted results to the BRATS 2013 evaluation platform [8]. It performed well on complete tumor (rank 2 on both data sets) and core (rank 2 and 3), but not as well on enhanced core (rank 9). The lower performance on enhanced core is not surprising, as we base the segmentation on one Gaussian without any prior to separate it from the rest of the core. Average Dice scores and robust Hausdorff distances (95% quantile) on all data sets are shown in table 1. The results on the 2015 training data set are lower, as it includes more difficult subjects with substantial artifacts, more progressed tumors and resections.

| | Dice [%] | | | | | | Hausdorff [mm] | | |
|---|---|---|---|---|---|---|---|---|---|
| Data sets | Comp., | *HG/LG* | Core, | *HG/LG* | Enh., | *HG/LG* | Comp. | Core | Enh. |
| 2015 Training | 77 ± 19 | *76/78* | 64 ± 29 | *69/44* | 52 ± 33 | *58/31* | 18 | 17 | 15 |
| 2013 Challenge | 87 ± 3 | *87/–* | 82 ± 15 | *82/–* | 70 ± 15 | *70/–* | – | – | – |
| 2013 Leaderb. | 83 ± 17 | *87/59* | 71 ± 27 | *78/32* | 54 ± 51 | *64/0* | – | – | – |

**Table 1.** Average Dice and Hausdorff scores. Hausdorff for enhanced core excludes 12 subjects due to missing label in either the ground truth or estimated segmentation.

**Acknowledgements:** This research was supported by NIH NCRR (P41-RR14075), NIBIB (R01EB013565) and the Lundbeck Foundation (R141-2013-13117).

## References

1. Lee, H., Grosse, R., Ranganath, R., Ng, A. Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning, ACM (2009)
2. Fischer, A., Igel, C.: Training restricted Boltzmann machines: An introduction. Pattern Recognition 47(1) (2014) 25-39
3. Melchior, J., Fischer, A., Wang, N., Wiskott, L.: How to Center Binary Restricted Boltzmann Machines. arXiv preprint arXiv:1311.1354 (2013)
4. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based tissue classification of MR images of the brain. IEEE Transactions on Medical Imaging 18(10) (1999)
5. Ashburner, J., Friston, K., Holmes, A., Poline, J.-B.: Statistical Parametric Mapping. The Wellcome Dept. Cognitive Neurology, Univ. College London, London, U.K. Available: http://www.fil.ion.ucl.ac.uk/spm/
6. Murphy, K. P.: Machine learning: a probabilistic perspective. MIT Press (2012)
7. Menze, B. H., et al.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). To appear in IEEE Transactions on Medical Imaging (2015)
8. Kistler, M., Bonaretti, S., Pfahrer, M., Niklaus, R., Büchler, P.: The virtual skeleton database: an open access repository for biomedical research and collaboration. Journal of Medical Internet Research 15(11) (2013)

Chapter 11

# Paper E

# An Ensemble of 2D Convolutional Neural Networks for Tumor Segmentation

Mark Lyksborg( ), Oula Puonti, Mikael Agn, and Rasmus Larsen

Department of Applied Mathematics and Computer Science,
Technical University of Denmark, Kgs. Lyngby, Denmark
{mlyk,oupu,miag,rlar}@dtu.dk

**Abstract.** Accurate tumor segmentation plays an important role in radiosurgery planning and the assessment of radiotherapy treatment efficacy. In this paper we propose a method combining an ensemble of 2D convolutional neural networks for doing a volumetric segmentation of magnetic resonance images. The segmentation is done in three steps; first the full tumor region, is segmented from the background by a voxel-wise merging of the decisions of three networks learned from three orthogonal planes, next the segmentation is refined using a cellular automaton-based seed growing method known as growcut. Finally, within-tumor sub-regions are segmented using an additional ensemble of networks trained for the task. We demonstrate the method on the MICCAI Brain Tumor Segmentation Challenge dataset of 2014, and show improved segmentation accuracy compared to an axially trained 2D network and an ensemble segmentation without growcut. We further obtain competitive Dice scores compared with the most recent tumor segmentation challenge.

**Keywords:** Tumor segmentation · Convolutional neural network · Ensemble classification · Cellular automaton

## 1 Introduction

Segmentation of brain tumors plays a role in radiosurgery, radiotherapy planning, and for monitoring tumor growth. Segmentation is challenging since tumor location and appearance vary greatly between patients.

Many successful method for doing voxel-based segmentation are based on the random forest (RF) classification scheme which predicts segmentation labels from user engineered image features. Tustison et al. [15] proposed a two-stage RF approach, with features derived from a Gaussian mixture model followed by a Markov random field segmentation smoothing. The RF was also used by Reza et al. [12] who designed features using textons and multifractional Brownian motion. Menze et al. [10] proposed a generative probabilistic atlas-based model which adapts to the intensity distribution of different subjects and later combined it with the RF classifier [9]. An example of a successfull method that does not use a RF classifier is the patch-based approach [2]. Here voxels are

© Springer International Publishing Switzerland 2015
R.R. Paulsen and K.S. Pedersen (Eds.): SCIA 2015, LNCS 9127, pp. 201–211, 2015.
DOI: 10.1007/978-3-319-19665-7_17

segmented by comparing image patches to a dictionary consisting of training patches where the corresponding expert labels are used for segmentation.

In recent years and due to advancements in computational power, deep neural networks have been revived. In the most recent Brain Tumor Segmentation Challenge 2014 (BraTS2014), this was reflected by a number of contributions using deep neural networks. The work by Davy et al. [3] presented a 2D convolutional network trained from an axial perspective. Two others presented 3D networks [16], [18], and while their implementations differed, the results indicated a benefit of using 3D information. An important property of a network is that it learns image features relevant for the specific segmentation problem. This alleviate researchers from having to engineer such features.

We revisit the idea of Davy et al. [3] but instead of using one 2D network to do voxel-based segmentations, we learn an ensemble of networks, one for each of the axial, sagittal and coronal planes and fuse their segmentations into a more accurate 3D informed segmentation. Unlike previous works using convolutional networks we do not segment the tumor and its sub-regions using a single multi-label classifier. Instead, we split the problem into two sequential segmentation problems. The first segmentation separates tumor from healthy tissue and refine the segmentation using a growcut algorithm [17]. The second segmentation performs the within-tumor sub-region segmentation using the tumor mask of the first segmentation to select voxels of interest.

The method (Fig. 1) is demonstrated on the BraTS2014 dataset. We were able to achieve improved ground truth segmentation accuracy compared to a 2D axially trained network [3] and Dice scores [4] just below the top methods of the challenge leaderboard (https://www.virtualskeleton.ch/BRATS/Start2014).

## 2 Data

Two datasets were downloaded from the BraTS2014 website (November, 2014).

The first dataset (data1) consisted of 106 high grade glioma (HGG) and 25 low grade glioma (LGG) subjects (no longitudinal repetitions), all with ground truth segmentations of the tumors. It was randomly split into a training set of 76 HGG/15 LGG subjects, and the rest (30 HGG/10 LGG) were used as test data. For each subject, we used a set of multimodal magnetic resonance imaging (MRI) volumes, consisting of two T2-weighted images (Fluid-attenuated inversion recovery (FLAIR) and (T2)) and a T1-weighted image with gadolinium contrast (T1c). The MRIs were skull stripped, rigidly oriented according to MNI space and re-sliced to 1 $mm^3$ as described in [6]. The ground truth segmentation consisted of five labels (background=0, necrosis=1, edema=2, non-enhancing=3, enhancing=4).

The second dataset (data2) consisted of 187 multi-modal MRI volumes from 88 different subjects with 99 longitudinal repetitions. Since only the BraTS2014 challenge organizers know the ground truth segmentations, it allowed for a blinded segmentation evaluation via the challenge website.

## 3 Method

The proposed method, outlined in Fig. 1, consists of four steps. First, the MRI volumes are bias corrected for scanner field inhomogeneity and standardized to similar cross subject intensities. Second, an ensemble of convolutional networks segments the tumor from healthy tissue. The third step (growcut) post processes the segmentation to improve the segmentation. The fourth step does the within-tumor segmentation using an additional ensemble of networks. The four steps of the method are detailed successively in section 3.1-3.4.

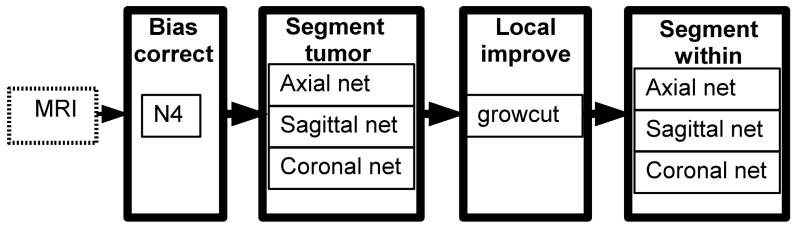



**Fig. 1.** Shows a schematic, outlining the pipeline of our method. The multi-modal MRI data is pushed through four successive stages of 1) bias correction, 2) whole tumor segmentation (tumor vs. none tumor), 3) localized post-processing of the segmentation and 4) a within-tumor segmentation stage.

### 3.1 Bias Correction and Standardization

MRI generally exhibits large intensity variations even within the same tissue type of a subject, largely due to field inhomogeneity of the scanner. To minimize this bias, the N4 method [14] was applied to each MRI.The N4 method works under the assumption that the bias field can be modeled by a smooth multiplicative model which is fitted iteratively to maximize the high frequency content of the MRI intensity distribution. To further standardize across different scanners, the maximum peak of each MRI intensity histogram was found, and the intensities scaled according to $I = I_c \cdot (I_b/I_p)$, where $I_c$ is the N4 bias corrected image volume, $I_p$ is the maximum peak intensity of $I_c$ and $I_b$ is a reference value which we fixed to $I_b = 200$. To achieve equal importance of the multi-modal MRI, their intensities were further standardised using a normal transformation applied to each of the different modalities.

### 3.2 Convolutional Network Ensemble: Whole Tumor

To segment tumor tissue, three convolutional neural networks were trained using a multi-modal image patch of dimension $46 \times 46$. Each 2D network learned to classify the same center voxel but viewed from an axial, sagittal and coronal perspective. Combining this ensemble of 2D networks enabled the segmentation method to become 3D aware.

The 2D networks are described by the architecture in Fig. 2. It shows a network consisting of 6 layers. Each perform an algebraic operation on the input data $x$ and passes the result as input to the next layer. The process is repeated until reaching layer 6 which predicts the most probable classification label.

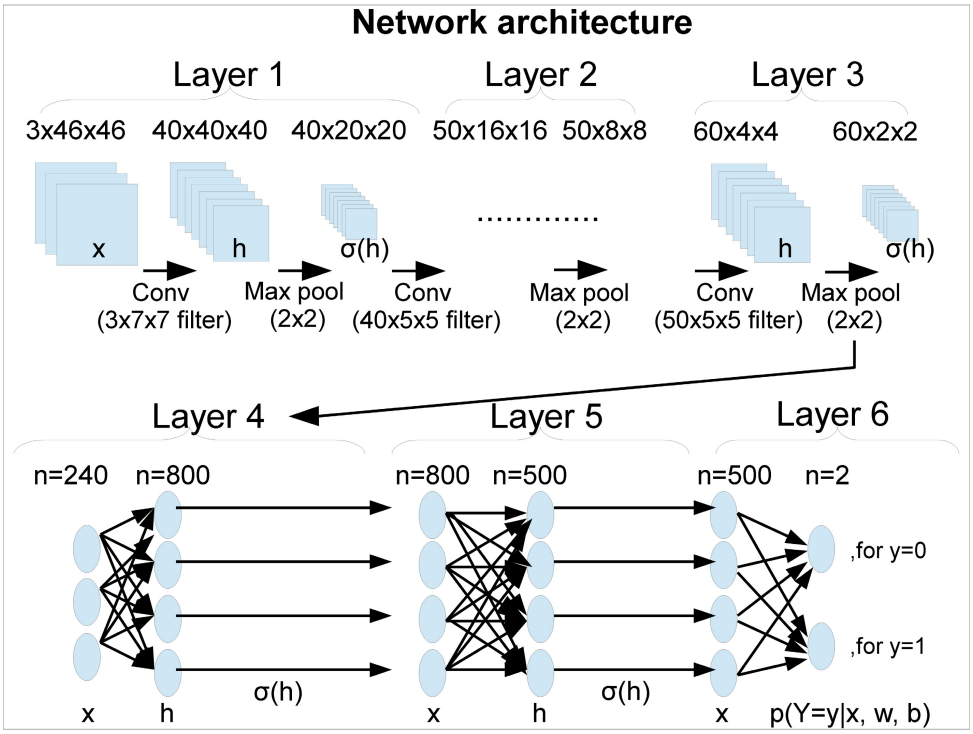



**Fig. 2.** Depicts a 2D deep neural network architecture consisting of six layers. The first three are convolutional layers, followed by two fully connected layers and a softmax layer where the arrows indicate the connections between layers. The squares illustrate the 2D nature of the input (x) and the intermediate representations (h) of the convolutional layers, where $x = [x_1...x_n]$ is a 3D matrix of $n$ input patches and $h = [h_1...h_m]$, is the concatenation of $m$ 2D filter response. The circles of the fully connected layers indicate its 1D nature with $n$ being the number of neurons (=the circles), such that $x = [x_1...x_n]^T$ and $h = [h_1...h_n]^T$ are the 1D vector representations of the input and the neuronal activations.

*Convolutional layers:* The convolutional layers apply filtering and downsampling operations to image patches. The first layer uses a filter bank of size $40 \times 3 \times 7 \times 7$ which it applies to the $3 \times 46 \times 46$ image patch. This produces a feature map $h$ of size $40 \times 40 \times 40$, where the first dimension indexes the feature maps, while the second and third dimensions indexes (row, column) coordinates. More specifically the $j^{th}$ map is calculated by $h_j = b_j + \sum_{i=1}^{n}(w_{ij} * x_i)$, where $i$

indexes the input channel and a trainable filter $w_{ij}$, the $*$ operator denotes 2D convolution and $n = 3$ is the number of input channels. Subsequently a $2 \times 2$ max pooling strategy is used to downsample $h$ to size $40 \times 20 \times 20$ and the rectified linear unit function, $\sigma(h) = max(0, h)$ is applied. The remaining convolutional layers (two and three) perform the same type of operations but using filter banks of size $50 \times 40 \times 5 \times 5$ and $60 \times 50 \times 5 \times 5$ for the respective layers. The application of these filters and downsampling steps result in a number of the intermediate feature maps with the dimensionalities listed in the top part of Fig. 2.

*Fully connected layers:* Layer 4, 5 and 6 are fully connected layers meaning each neuron is exposed to the full input $x$ of the previoues layer. Each of the 800 neurons in layer 4, evaluates the product $h_j = w_j^T x + b_j$ and applies the non-linear activation function $\sigma(h_j)$. Thereby transforming the 240 dimensional vector $x$ into an 800 dimensional vector $\sigma(h)$ which is passed to layer 5. Layer 5 works similar to layer 4, but now generating a 500 dimensional feature vector $\sigma(h)$ which is propagated to layer 6. Layer 6 evaluates the softmax function

$$p(Y = y|x, w, b) = \frac{e^{w_y x + b_y}}{\sum_j e^{w_j x + b_j}}, \tag{1}$$

generating posterior probabilities for a number of classification labels, $y = \{0, 1\}$. Here $w_j$ refer to a vector of linear parameters for the $j^{th}$ class, $b_j$ is a bias weight and x is the 500 dimensional response vector from the previous layer.

**Network Training** Each of the 2D networks were trained by minimizing the following cost function

$$C(W, B) = \frac{1}{nd} \cdot \sum_{i=1}^{nd} -\ln(p(Y = y^i|x^i, W, B)) + \lambda \cdot \sum_{j=1}^{nw} W_j^2. \tag{2}$$

The first term of eq. (2) is the mean negative log-likelihood of the softmax probability and we have used capitalized $(W, B)$ to indicate that it is a function of $(w, b)$ parameters from different types of layers. Further, the training patches are denoted $x^i, y^i$, corresponding to the patch intensities and ground truth label of the $i^{th}$ training example. The second term of eq. (2) is a regularization term that adds robustness to the optimization problem by limiting the solution space to models with smaller parameter weights. It does so by penalizing the 2-norm of the parameters and through experimentation we found $\lambda = 0.0001$ to be suitable.

The cost function was minimized using a stochastic gradient descent (SGD) which relied on the back propagation algorithm to estimate gradients. The SGD performed iterative updates based on gradients estimated from mini-batches with a batch size of 200 where an update occurred after each mini-batch. Each gradient update was further augmented by a moment based learning rule [13] which updated the parameters as a weighted combination of the current gradients and the gradients of previous iteration update. We used a momentum coefficient of 0.9. Layer 4 and 5 were trained using the dropout learning [5]

(dropout rate=0.5) which activates half the neurons for each training example. As a consequences the activations of these layers($\sigma(h)$) were divided by 2 when a network was applied to an unseen test image patch.

A GPU implementation for training the three 2D networks was achived using Theano [1].

**Network Ensemble Merging** Having learned the parameters of the three networks, their complementary decision information were merged. This was done using the posterior probablities of the last layer (layer 6). If the networks agreed on the same label we were highly confident in this classification and assigned the label of voxel x with probability $p(Y|x) = 1$. Otherwise a majority vote decided the class label and the probability was set to reflect this uncertainty by averaging the class probabilities of the three networks, $p(Y|x) = (1/3)\sum_{i=1}^{3} p_i(Y|x, w, b)$. The resulting label segmentations and their probabilities were then used as input for the growcut algorithm.

### 3.3 Cellular Automaton: Growcut

The growcut algorithm was initially proposed as a continuous state cellular automata method for automated segmentation based on user labeled seed voxels [17]. From these labels and a local intensity transition rule the algorithm decides whether voxels should be re-labelled.

We used the algorithmic formulation of [17] which we extended to 3D. The algorithm models each voxel as a cell with a state set $S(\Theta, l, C)$ consisting of a strength value $\Theta \in [0, 1]$, a label $l$ and an intensity feature vector $C$. It is an iterative algorithm and for each iteration the strength and labels of the previous iteration remain fixed. During an iteration each image cell $r$ is attacked by its neighboring cells $s \in N(r)$ where $N(r)$ denote the $3 \times 3 \times 3$ neighborhood of a volume and only if $g(C_r, C_s) \cdot \Theta_s > \Theta_r$, will $\Theta_r$, and $l_r$ be updated before the next iteration. The local transition rule is given by

$$g(C_1, C_2) = 1 - \frac{||C_1 - C_2||_2}{k} \tag{3}$$

Where we have normalized the intensities of C to be in the range [0, 1] such that for $k = \sqrt{3}$, the value of $g(C_1, C_2) \in [0, 1]$. Since $g(C_1, C_2)$ can never exceed 1, any cells with strength $\Theta = 1$ will remain constant throughout the algorithm.

To use the growcut on the ensemble segmentations, the feature vector $C$ was set to the multi-modal MRI intensities and the values of $l$, $\Theta$ were initialized with the labels and probability maps of the convolutional network ensemble. This initialization served as a strong prior for growcut segmentation, assuming that the segmentation was already near optimal.

Once growcut converged to a stable segmentation (100 iterations), a heuristic rule was used to identify the tumor. It was based on a connected components analysis to remove any spatially coherent clusters of voxels which were less than 80% of the biggest cluster.

### 3.4 Convolutional Network Ensemble: Within-Tumor

This ensemble of convolutional networks was used to segment the within-tumor sub-regions. The architecture of each network is similar to the previously described, but considers a smaller image patch and has only two convolutional layers, two fully connected dropout layers and softmax probability layer. The input patch size is $3 \times 34 \times 34$ and the first convolutional layer uses a filter bank of size $50 \times 3 \times 7 \times 7$ while the second one uses a filter bank of size $60 \times 50 \times 5 \times 5$. The justification of choosing a smaller patch size is that the within-tumor segmentation uses information on a smaller scale compared to the whole tumor segmentation. As with the previously described networks, the fully connected layers use 800 and 500 neurons respectively while the softmax layer, predicts one of four possible classification labels. The SGD optimization was again used to train the networks but for these specific networks we used $\lambda = 0.00005$.

**Network Ensemble Merging** The voxel-based decisions of the ensemble of axial, sagittal and coronal networks were either set to the label they all agree on, or according to the most probable average probability of the softmax probability.

## 4 Results

### 4.1 Test and Phenotype Performance

Testing our method on the 40 left out subjects (data1), resulted in the segmentation performances of Table 1. This table shows ground truth scores for three methods; A 2D convolutional network applied to the axial plane similar to [3], a method using only the ensemble part of our method (ensem) and our full method which is ensem in combination with growcut (ensem+grow). The scores of the table are given for pathologically relevant tumor regions. These are the whole tumor (labels: necrosis, edema, non-enhancing, enhancing), the enhanced tumor region and the tumor core (labels: necrosis, non-enhancing, enhancing). We see that using an ensemble improved the segmentation relative to a 2D network and achieved further improvement by including growcut post-processing. As a visual comparison example, two tumor segmentations based on our method and their

**Table 1.** Average segmentation performance scores of three convolutional neural network methods evaluated on 40 subjects of data1. The scores (Dice, positive predictive and sensitivity) were calculated for the different tumor regions.

| Method | Dice scores | | | Positive predictive | | | Sensitivity | | |
|---|---|---|---|---|---|---|---|---|---|
| | Whole | Core | Enh. | Whole | Core | Enh. | Whole | Core | Enh. |
| axial | 0.744 | 0.642 | 0.629 | 0.732 | 0.624 | 0.642 | 0.811 | 0.746 | 0.707 |
| ensem | 0.786 | 0.686 | 0.676 | 0.786 | 0.707 | 0.693 | 0.825 | 0.743 | 0.717 |
| ensem+grow | 0.810 | 0.697 | 0.681 | 0.833 | 0.718 | 0.701 | 0.825 | 0.750 | 0.720 |

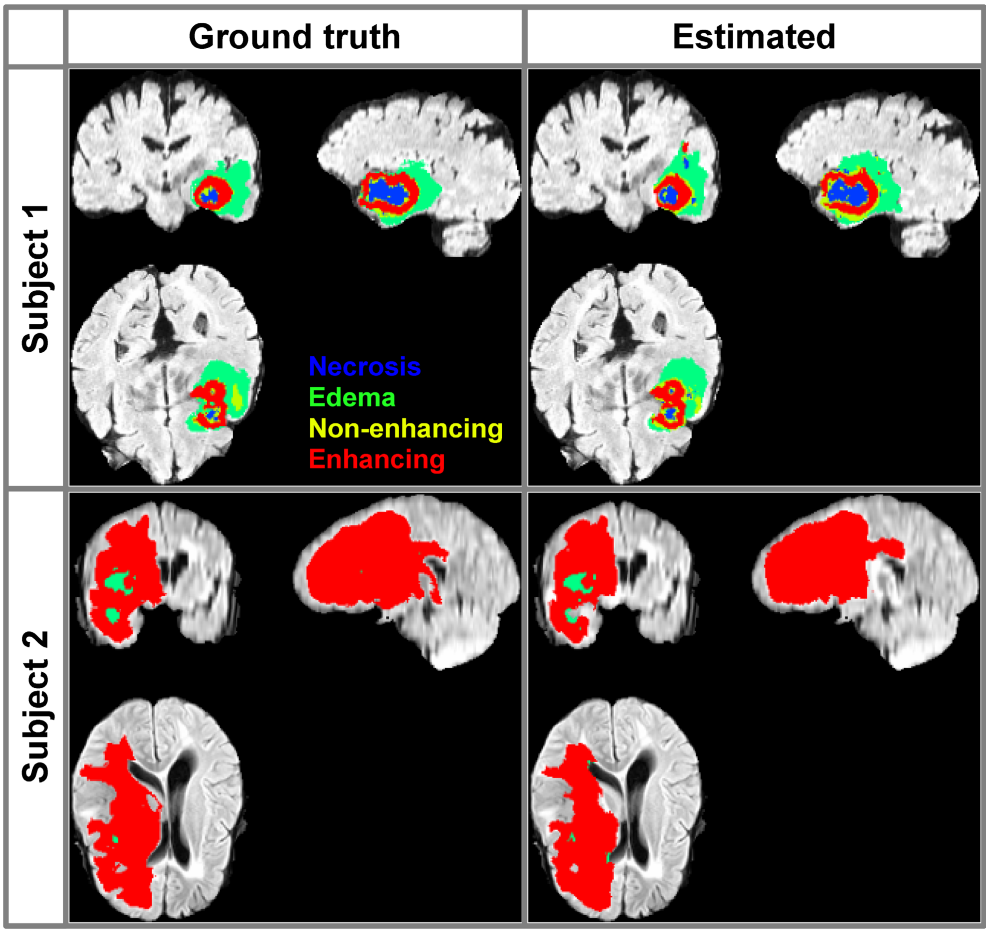



**Fig. 3.** This visual comparison shows both the proposed segmentation method and corresponding ground truth for two subjects. The Dice scores of subject 1 were 0.825 (whole), 0.795 (core) and 0.842 (enhanced) and for subject 2 they were, 0.892 (whole), 0.840 (core) and 0.854 (enhanced).

ground truth, are shown in Fig. 3. By dividing the test subjects based on tumor types (HGG/LGG), we evaluated their impact on method performance. This comparison (Fig. 4), reveals higher Dice scores with less variance for the HGGs, indicating a methodological bias towards the tumor type.

### 4.2 Blinded Challenge Performance

Testing our method on the blinded challenge dataset previously denoted data2 and performing an on-line evaluation of the segmentations, resulted in the average performance scores of Table 2. It lists the scores for the first time point of the 99 subjects (cross sectional) and the full challenge data (full data) where similar performances are achieved. It also includes the top 3 scores of the BraTS2014 challenge where our method is ranked amongst.

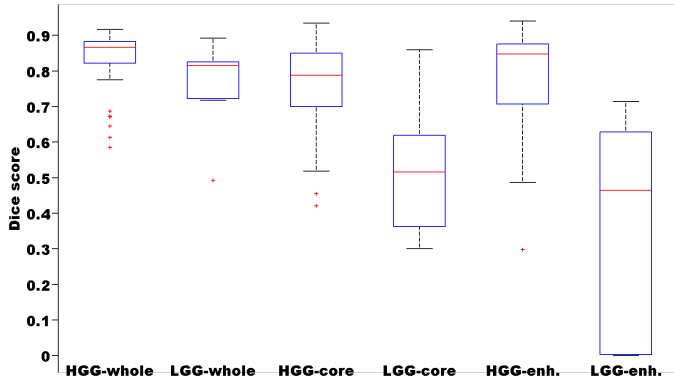



**Fig. 4.** Ground truth Dice scores performance for two different types of tumors (HGG and LGG). Red line indicate mean Dice score, blue boxes show the 25 and 75 percentiles of the scores while extreme observations are show with red dots.

**Table 2.** Shows the average segmentation performance scores of our method in grey (cross sectional and full data), for the BraTS2014 challenge data (data2). Also listed are the top three of the challenge (15/12-2014), ranked according to their whole tumor Dice scores. These are Urbag [16], Kleej [7], Dvorp [8].

| Method | Dice scores | | | Positive predictive | | | Sensitivity | | |
|---|---|---|---|---|---|---|---|---|---|
| | Whole | Core | Enh. | Whole | Core | Enh. | Whole | Core | Enh. |
| Cross sectional | 0.801 | 0.637 | 0.586 | 0.803 | 0.682 | 0.554 | 0.857 | 0.715 | 0.745 |
| Full data | 0.799 | 0.631 | 0.625 | 0.783 | 0.629 | 0.580 | 0.861 | 0.736 | 0.776 |
| Urbag | 0.87 | 0.76 | 0.72 | 0.91 | 0.80 | 0.69 | 0.85 | 0.76 | 0.81 |
| Kleej | 0.87 | 0.76 | 0.73 | 0.90 | 0.73 | 0.66 | 0.85 | 0.83 | 0.87 |
| Dvorp | 0.60 | 0.30 | 0.29 | 0.86 | 0.58 | 0.56 | 0.53 | 0.27 | 0.28 |

## 5 Discussion

We have presented a method, combining an ensemble of 2D convolutional networks with the growcut method for making a 3D informed segmentation. It showed improved accuracy compared to a 2D network and an ensemble segmentation without growcut thereby validating the usefulness of the proposed method. The investigation of tumor type showed better performance for HGG, likely due to the imbalanced training data distribution (76 HGG/15 LGG). It could also indicate the presence of a measurable pathologic difference. If so, the training of a segmentation method for each type could lead to improved segmentations for both types. This would require knowing the tumor type in advance, information that was not readily available for the blinded challenge data. Our challenge results showed a nice performance although sub-par to the top two methods of the challenge but was superior to the remaining 11. It is noted that our methods performance is in the Dice score range that manual annotators can achieve according the results of [11]. They reported the Dice accuracy of

annotators to be in the range of (0.74-0.85). This is comparable to the proposed method. A simple strategy for improving our work would be to extend the ensemble to use 3D network (computationally costly) or to investigate the inclusion of networks trained from more than orthogonal planes. In addition, the usage of using longitudinal information could also play a role towards improving segmentations.

## References

1. Bergstra, J., et al.: Theano: a CPU and GPU math expression compiler. In: Python for Scientific Computing Conference (SciPy) (2010)
2. Cordier, N., Menze, B., Delingette, H., Ayache, N.: Patch-based segmentation of brain tissues. In: MICCAI-BraTS (Challenge on Multimodal Brain Tumor Segmentation), pp. 6–17 (2013)
3. Davy, A., Havaei, M., Warde-Farley, D., Biard, A., Tran, L., Jodoin, P.M., Courville, A., Larochelle, H., Pal, C., Bengio, Y.: Brain tumor segmentation with deep neural networks. In: MICCAI-BraTS, pp. 1–5 (2014)
4. Dice, L.R.: Measures of the amount of ecologic association between species. Ecology (1945)
5. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Improving neural networks by preventing co-adaptation of feature detectors. CoRR (2012)
6. Jakab, A.: Segmenting brain tumors with the slicer 3d software. Tech. rep., University of Debrecen / ETH Zürich (2012)
7. Kleesiek, J., Biller, A., Urban, G., Kothe, U., Bendszus, M., Hamprecht, F.: Ilastik for multi-modal brain tumor segmentation. In: MICCAI-BraTS, pp. 12–17 (2014)
8. Kwon, D., Akbari, H., Da, X., Gaonkar, B., Davatzikos, C.: Multimodal brain tumor image segmentation using glistr. In: MICCAI-BraTS, pp. 18–19 (2014)
9. Menze, B., Geremia, E., Ayache, N., Szekely, G.: Segmenting glioma in multimodal images using a generative-discriminative model for brain lesion segmentation. In: MICCAI-BraTS, pp. 56–63 (2012)
10. Menze, B., Leemput, K.V., Lashkar, D., Weber, M., Ayache, N., Golland, P.: Segmenting glioma in multi-modal images using a generative model for brain lesion segmentation, pp. 49–55 (2012)
11. Menze, B.H., et al.: The multimodal brain tumor image segmentation benchmark (brats). IEEE Transactions on Medical Imaging (2014)
12. Reza, S., Iftekharuddin, K.: Improved brain tumor tissue segmentation using texture features. In: MICCAI-BraTS, pp. 27–30 (2014)
13. Sutskever, I., Martens, J., Dahl, G.E., Hinton, G.E.: On the importance of initialization and momentum in deep learning. In: 30th International Conference on Machine Learning (ICML 2013), vol. 28, pp. 1139–1147, May 2013
14. Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C.: N4ITK: Improved N3 Bias Correction. IEEE Trans. Med. Imaging **29**(6), 1310–1320 (2010)
15. Tustison, N., Wintermark, M., Durst, C., Avants, B.: Ants and arboles. In: MICCAI-BraTS, pp. 47–50 (2013)

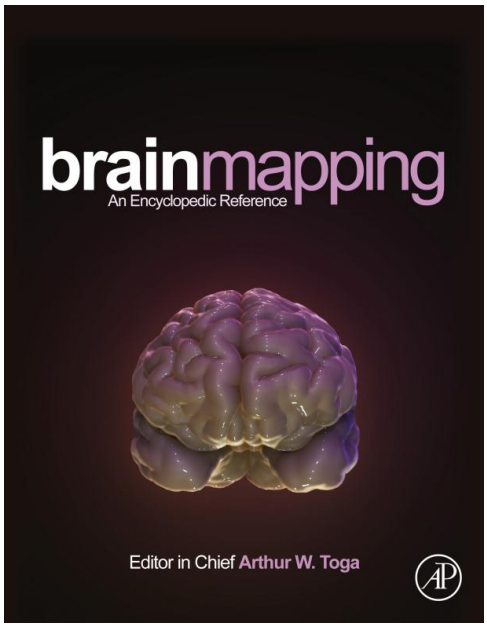
16. Urban, G., Bendszus, M., Hamprecht, F.A., Kleesiek, J.: Multi-modal brain tumor segmentation using deep convolutional neural networks. In: MICCAI-BraTSs, pp. 31–35 (2014)
17. Vezhnevets, V., Konouchine, V.: GrowCut - interactive multi-label n-d image segmentation by cellular automata. In: Proceedings of Graphicon (2005)
18. Zikic, D., Ioannou, Y., Brown, M., Criminisi, A.: Segmentation of brain tumor tissues with convolutional neural networks. In: MICCAI-BraTS, pp. 36–39 (2014)

Chapter 12

# Paper F

Provided for non-commercial research and educational use.
Not for reproduction, distribution or commercial use.

This article was originally published in Brain Mapping: An Encyclopedic Reference, published by Elsevier, and the attached copy is provided by Elsevier for the author's benefit and for the benefit of the author's institution, for non-commercial research and educational use including without limitation use in instruction at your institution, sending it to specific colleagues who you know, and providing a copy to your institution's administrator.




All other uses, reproduction and distribution, including without limitation commercial reprints, selling or licensing copies or access, or posting on open internet sites, your personal or institution's website or repository, are prohibited. For exceptions, permission may be sought for such use through Elsevier's permissions site at:

http://www.elsevier.com/locate/permissionusematerial

Van Leemput K., and Puonti O. (2015) Tissue Classification. In: Arthur W. Toga, editor. *Brain Mapping: An Encyclopedic Reference*, vol. 1, pp. 373-381. Academic Press: Elsevier.

Author's personal copy

# Tissue Classification

**K Van Leemput**, Harvard Medical School, Boston, MA, USA
**O Puonti**, Technical University of Denmark, Lyngby, Denmark

© 2015 Elsevier Inc. All rights reserved.

**Abbreviations**

EM Expectation-maximization
MAP Maximum a posteriori
ML Maximum likelihood

Computational methods for automatically segmenting magnetic resonance (MR) images of the brain have seen tremendous advances in recent years. So-called tissue classification techniques, aimed at extracting the three main brain tissue classes (white matter, gray matter, and cerebrospinal fluid), are now well established. In their simplest form, these methods classify voxels independently based on their intensity alone, although much more sophisticated models are typically used in practice (Anbeek, Vincken, van Bochove, van Osch, & van der Grond, 2005; Ashburner & Friston, 1997, 2005; Awate, Tasdizen, Foster, & Whitaker, 2006; Greenspan, Ruf, & Goldberger, 2006; Marroquin, Vemuri, Botello, Calderon, & Fernandez-Bouzas, 2002; Pham & Prince, 1999; Rajapakse, Giedd, & Rapoport, 1997; Van Leemput, Maes, Vandermeulen, & Suetens, 1999a,1999b; Warfield, Kaus, Jolesz, & Kikinis, 2000; Wells, Grimson, Kikinis, & Jolesz, 1996; Zeng, Staib, Schultz, & Duncan, 1999; Zhang, Brady, & Smith, 2001).

This article aims to give an overview of often-used computational techniques for brain tissue classification. Although other methods exist, we will concentrate on Bayesian modeling approaches, in which generative image models are constructed and subsequently 'inverted' to obtain automated segmentations. This general framework encompasses a large number of segmentation methods, including those implemented in widely used software packages such as SPM, FSL, and FreeSurfer, as well as techniques for automatically segmenting many more brain structures than merely the three main brain tissue types only (Ashburner & Friston, 2005; Fischl et al., 2002; Fischl, Salat et al., 2004; Fischl, van der Kouwe et al., 2004; Guillemaud & Brady, 1997; Held et al., 1997; Lorenzo-Valdes, Sanchez-Ortiz, Mohiaddin, & Rueckert, 2004; Marroquin et al., 2002; Menze et al., 2010; Pohl, Fisher, Grimson, Kikinis, & Wells, 2006; Pohl et al., 2007; Prastawa, Bullitt, Ho, & Gerig, 2004; Sabuncu, Yeo, Van Leemput, Fischl, & Golland, 2010; Van Leemput, Maes, Vandermeulen, Colchester, & Suetens, 2001; Van Leemput et al., 1999b; Wells et al., 1996; Xue et al., 2007; Zhang et al., 2001).

We first introduce the general modeling framework and the specific case of the Gaussian mixture model. We then discuss maximum likelihood (ML) parameter estimation and the expectation–maximization (EM) algorithm and conclude the article with further model extensions such as MR bias field models and probabilistic atlases.

## Generative Modeling Framework

Brain MR segmentation methods are often based on so-called generative models, that is, probabilistic models that describe how images can be generated synthetically. Such models generally consist of two parts:

- A segmentation prior that makes predictions about where neuroanatomical structures typically occur throughout the image. Let $\boldsymbol{l}=(l_1,\ldots,l_I)^{\mathrm{T}}$ be a (vectorized) label image with a total of $I$ voxels, with $l_i\in\{1,\ldots,K\}$ denoting the one of $K$ possible labels assigned to voxel $i$, indicating which of the $K$ anatomical structures the voxel belongs to. For the purpose of tissue classification, there are typically $K=3$ labels, namely, white matter, gray matter, and cerebrospinal fluid. The segmentation prior then consists of a probability distribution $p(\boldsymbol{l}|\boldsymbol{\theta}_l)$ that typically depends on a set of parameters $\boldsymbol{\theta}_l$.
- A *likelihood* function that predicts how any given label image, where each voxel is assigned a unique anatomical label, translates into an image where each voxel has an intensity. This is essentially a (often very simplistic) model of how an MR scanner generates images from known anatomy: given a label image $\boldsymbol{l}$, a corresponding intensity image $\boldsymbol{d}=(d_1,\ldots,d_I)^{\mathrm{T}}$ is obtained by random sampling from some probability distribution $p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{\theta}_d)$ with parameters $\boldsymbol{\theta}_d$, where $d_i$ denotes the MR intensity in voxel $i$.

In summary, the generative model is fully specified by two distributions $p(\boldsymbol{l}|\boldsymbol{\theta}_l)$ and $p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{\theta}_d)$, which often depend on parameters $\boldsymbol{\theta}=(\boldsymbol{\theta}_l^{\mathrm{T}},\boldsymbol{\theta}_d^{\mathrm{T}})^{\mathrm{T}}$ that are either assumed to be known in advance or, more frequently, need to be estimated from the image data itself. The exact form of the used distributions depends on the segmentation problem at hand. In general, the more realistic the models, the better the segmentations that can be obtained with them.

Once the exact generative model has been chosen and appropriate values $\hat{\boldsymbol{\theta}}$ for its parameters are known, properties of the underlying segmentation of an image can be inferred by inspecting the posterior probability distribution $p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right)$. Using Bayes' rule, this distribution is given by

$$p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right)=\frac{p\left(\boldsymbol{d}|\boldsymbol{l},\hat{\boldsymbol{\theta}}_d\right)p\left(\boldsymbol{l}|\hat{\boldsymbol{\theta}}_l\right)}{p\left(\boldsymbol{d}|\hat{\boldsymbol{\theta}}\right)} \quad [1]$$

Author's personal copy



with $p\left(\boldsymbol{d}|\hat{\boldsymbol{\theta}}\right)=\sum_l p\left(\boldsymbol{d}|\boldsymbol{l},\hat{\boldsymbol{\theta}}_d\right)p\left(\boldsymbol{l}|\hat{\boldsymbol{\theta}}_l\right)$. For instance, one might look for the segmentation $\hat{\boldsymbol{l}}$ that has the maximum a posteriori (MAP) probability

$$\hat{\boldsymbol{l}}=\arg\max_{\boldsymbol{l}} p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right) \quad [2]$$

or estimate the volume of the anatomical structure corresponding to label $k$ by assessing its expected value

$$\sum_{\boldsymbol{l}} V_k(\boldsymbol{l})p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right) \quad [3]$$

where $V_k(\boldsymbol{l})$ counts the number of voxels that have label $k$ in $\boldsymbol{l}$.

## Gaussian Mixture Model

A very simple generative model that is nevertheless quite useful in practice is the so-called Gaussian mixture model. In this model, the segmentation prior is of the form

$$p(\boldsymbol{l}|\boldsymbol{\theta}_l)=\prod_i p(l_i|\boldsymbol{\theta}_l) \quad [4]$$

$$p(\boldsymbol{l}|\boldsymbol{\theta}_l)=\prod_i \pi_{l_i} \quad [5]$$

where the parameters $\boldsymbol{\theta}_l=(\pi_1,\ldots,\pi_K)^{\mathrm{T}}$ consist of a set of probabilities $\pi_k$ satisfying $\pi_k\geq 0,\forall k$ and $\sum_k\pi_k=1$. In other words, this model assumes that the labels are assigned to the voxels independently from one another, that is, the probability that a certain label occurs in a particular voxel is unaffected by the labels assigned to other voxels (eqn [4]) and each label occurs, on average, with a relative frequency of $\pi_k$ (eqn [5]).

For the likelihood function, it is assumed that the intensity in each voxel only depends on the label in that voxel and not on that in other voxels

$$p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{\theta}_d)=\prod_i p(d_i|l_i,\boldsymbol{\theta}_d) \quad [6]$$

and that the intensity distribution associated with each label $k$ is Gaussian with mean $\mu_k$ and variance $\sigma_k^2$:

$$p(d_i|l_i,\boldsymbol{\theta}_d)=\mathcal{N}\left(d_i|\mu_{l_i},\sigma_{l_i}^2\right) \quad [7]$$

where

$$\mathcal{N}(d|\mu,\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(d-\mu)^2}{2\sigma^2}\right] \quad [8]$$

and $\boldsymbol{\theta}_d=(\mu_1,\ldots,\mu_K,\sigma_1^2,\ldots\sigma_K^2)^{\mathrm{T}}$.

It is instructive to write down the probability with which this model generates a given image $\boldsymbol{d}$:

$$\begin{aligned}p(\boldsymbol{d}|\boldsymbol{\theta})&=\sum_{\boldsymbol{l}} p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{\theta}_d)p(\boldsymbol{l}|\boldsymbol{\theta}_l)\\&=\sum_{\boldsymbol{l}}\left[\prod_i \mathcal{N}\left(d_i|\mu_{l_i},\sigma_{l_i}^2\right)\prod_i \pi_{l_i}\right]=\prod_i p(d_i|\boldsymbol{\theta})\end{aligned} \quad [9]$$

with

$$p(d|\boldsymbol{\theta})=\sum_k \mathcal{N}\left(d|\mu_k,\sigma_k^2\right)\pi_k \quad [10]$$

Equation [10] explains why this model is called the Gaussian mixture model: the intensity distribution in any voxel, independent of its spatial location, is given by the same linear superposition of Gaussians. Since no spatial information is encoded in the model, it can directly be visualized as a way to approximate the histogram, as shown in Figure 1.

Because of the assumption of statistical independence between voxels, the segmentation posterior (eqn [1]) reduces to a simple form that is factorized (i.e., appears as a product) over the voxels:

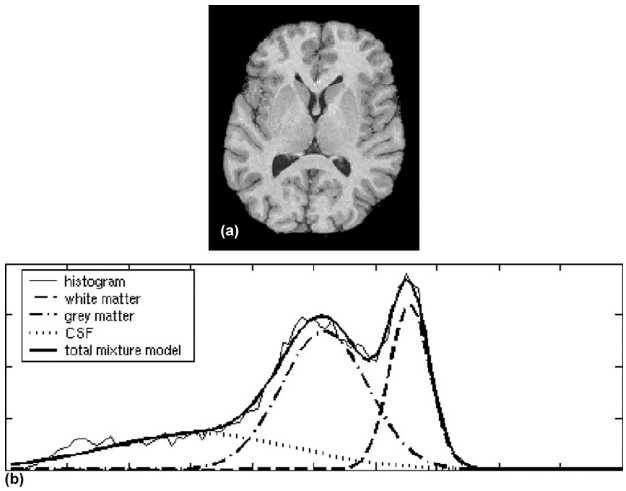



**Figure 1** In the Gaussian mixture model, the histogram is described as a linear superposition of Gaussian distributions: (a) MR scan of the head, after removing all non-brain tissue and other pre-processing steps; and (b) corresponding histogram and its representation as a sum of Gaussians.

Author's personal copy



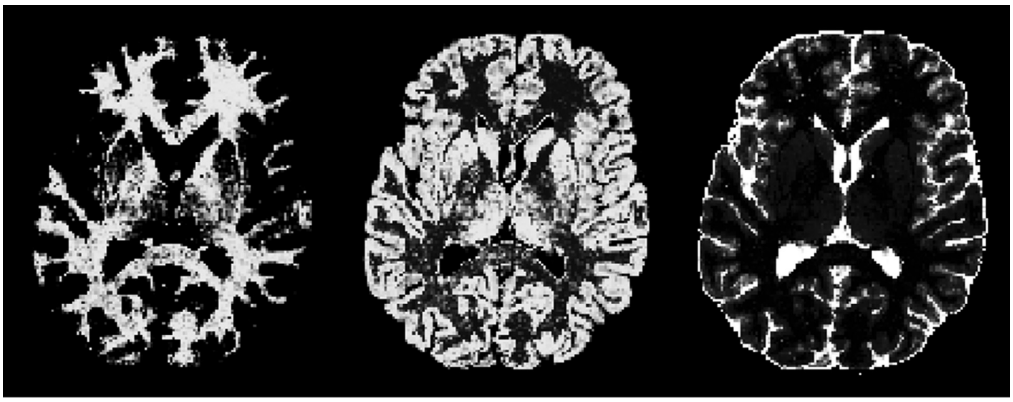


**Figure 2** Visualization of the segmentation posterior corresponding to the data and model of figure 1. High and low intensities correspond to high and low probabilities, respectively.

$$p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right)=\frac{p\left(\boldsymbol{d}|\boldsymbol{l},\hat{\boldsymbol{\theta}}_d\right)p\left(\boldsymbol{l}|\hat{\boldsymbol{\theta}}_l\right)}{p\left(\boldsymbol{d}|\hat{\boldsymbol{\theta}}\right)}=\frac{\prod_i\mathcal{N}\left(d_i|\hat{\mu}_{l_i},\hat{\sigma}_{l_i}^2\right)\prod_i\hat{\pi}_{l_i}}{\prod_i\sum_k\mathcal{N}\left(d_i|\hat{\mu}_k,\hat{\sigma}_k^2\right)\hat{\pi}_k}$$
$$=\prod_i p\left(l_i|d_i,\hat{\boldsymbol{\theta}}\right) \quad [11]$$

where

$$p\left(l_i|d_i,\hat{\boldsymbol{\theta}}\right)=\frac{\mathcal{N}\left(d_i|\hat{\mu}_{l_i},\hat{\sigma}_{l_i}^2\right)\hat{\pi}_{l_i}}{\sum_k\mathcal{N}\left(d_i|\hat{\mu}_k,\hat{\sigma}_k^2\right)\hat{\pi}_k} \quad [12]$$

Therefore, the segmentation posterior is fully specified by each voxel's $k$ posterior probabilities of belonging to each structure; such segmentation posteriors can be visualized as images where high and low intensities correspond to high and low probabilities, respectively. The segmentation corresponding to the image and Gaussian mixture model of Figure 1 is visualized in Figure 2 this way. It is worth noting that the sum of all the structures' posterior probabilities adds to one in each voxel: $\sum_k p\left(k|d_i,\hat{\boldsymbol{\theta}}\right)=1,\ \forall i$.

Because of the factorized form of the segmentation posterior, the MAP segmentation (eqn [2]) is simply given by

$$\hat{\boldsymbol{l}}=\arg\max_{\boldsymbol{l}} p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right)=\arg\max_{l_1,\ldots,l_I} p\left(l_i|d_i,\hat{\boldsymbol{\theta}}\right) \quad [13]$$

that is, each voxel is assigned exclusively to the label with the highest posterior probability. Similarly, the expected volume of the anatomical structure corresponding to label $k$ is given by (eqn [3])

$$\sum_{\boldsymbol{l}} V_k(\boldsymbol{l})p\left(\boldsymbol{l}|\boldsymbol{d},\hat{\boldsymbol{\theta}}\right)=\sum_i p\left(k|d_i,\hat{\boldsymbol{\theta}}\right) \quad [14]$$

that is, a 'soft' count of voxels belonging to the structure, where voxels contribute according to their posterior probability of belonging to that structure.

## Parameter Optimization Using the EM Algorithm

So far, we have assumed that appropriate values $\hat{\boldsymbol{\theta}}$ of the model parameters are known in advance. One possible strategy to estimate these parameters is to manually click on some representative points in the image to be segmented – or in similar images obtained from other subjects – and then collect statistics on the intensity of the selected voxels. In general, however, such a strategy is cumbersome for such a versatile imaging modality as MR, where intensities do not directly correspond to physical properties of the tissue being scanned. By merely altering the imaging protocol, upgrading the scanner, or collecting images from different scanner models or manufacturers, the values of $\hat{\boldsymbol{\theta}}$ become inappropriate and need to be constructed again using manual interaction.

This difficulty can be avoided by estimating appropriate values for the model parameters automatically from each individual scan. This can be accomplished by estimating the parameters that maximize the so-called likelihood function $p(\boldsymbol{d}|\boldsymbol{\theta})$, which expresses how probable the observed image $\boldsymbol{d}$ is for different settings of the parameter vector $\boldsymbol{\theta}$:

$$\hat{\boldsymbol{\theta}}=\arg\max_{\boldsymbol{\theta}}\left[p(\boldsymbol{d}|\boldsymbol{\theta})\right]=\arg\max_{\boldsymbol{\theta}}\left[\log p(\boldsymbol{d}|\boldsymbol{\theta})\right] \quad [15]$$

The last step is true because the logarithm is a monotonically increasing function of its argument; it is used here because it simplifies the subsequent mathematical analysis and also avoids numerical underflow problems in practical computer implementations. The parameter vector $\hat{\boldsymbol{\theta}}$ resulting from eqn [15] is commonly called the maximum likelihood (ML) parameter estimate.

Maximizing the (log) likelihood function in image segmentation problems is a nontrivial optimization problem for which iterative numerical algorithms are needed. Although a variety of standard optimization methods could potentially be used, for the Gaussian mixture model, a dedicated and highly effective optimizer is available in the form of the so-called expectation–maximization algorithm (EM). The EM algorithm belongs to a family of optimization methods that work by repeatedly constructing a lower bound to the objective function, maximizing that lower bound, and repeating the process until convergence (Hunter & Lange, 2004). This process is illustrated in Figure 3. For a given starting estimate of the model parameters $\tilde{\boldsymbol{\theta}}$, a function of the model parameters $Q\left(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}}\right)$ is constructed that equals the log likelihood function at $\tilde{\boldsymbol{\theta}}$

Author's personal copy



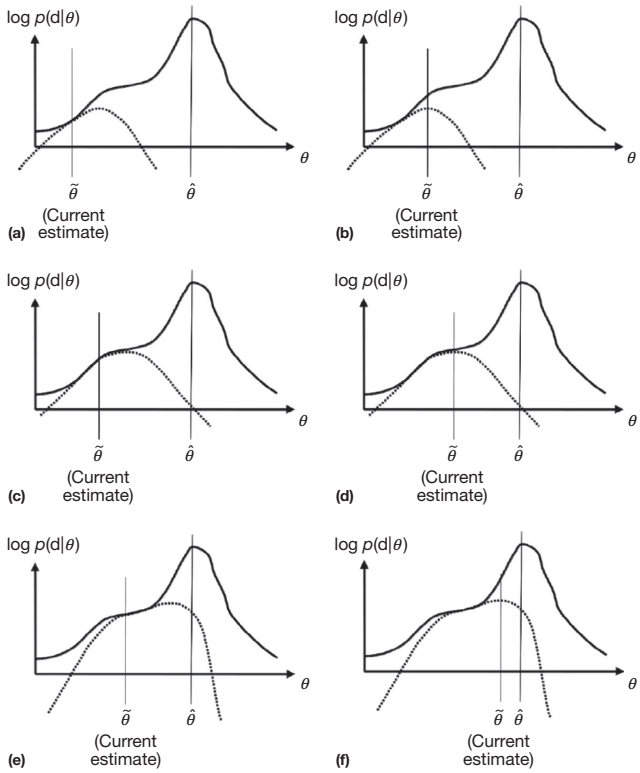



**Figure 3** In the EM algorithm the maximum likelihood parameters are sought by repeatedly constructing a lower bound to the log likelihood function, in such a way that the lower bound touches the log likelihood function exactly at the current parameter estimate (a). Subsequently the parameter estimate is updated to the parameter vector that maximizes the lower bound (b). A new lower bound is then constructed at this new location (c) and maximized again (d), and so forth ((e) and (f)), until convergence. In these plots, the log likelihood function is represented by a full line, and the successive lower bounds with a broken line.

$$Q(\tilde{\boldsymbol{\theta}}|\tilde{\boldsymbol{\theta}}) = \log p(\boldsymbol{d}|\tilde{\boldsymbol{\theta}}) \quad [16]$$

but that otherwise never exceeds it

$$Q(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}}) \leq \log p(\boldsymbol{d}|\boldsymbol{\theta}), \forall \boldsymbol{\theta} \quad [17]$$

The parameter vector maximizing $Q(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}})$ is then computed and used as the new parameter estimate $\tilde{\boldsymbol{\theta}}$, after which the whole process is repeated. Critically, because of eqns [16] and [17], updating the estimate $\tilde{\boldsymbol{\theta}}$ to the parameter vector that maximizes the lower bound automatically guarantees that the log likelihood function increases, by at least the same amount as the lower bound has increased. The consecutive estimates $\tilde{\boldsymbol{\theta}}$ obtained this way are therefore increasingly better estimates of the ML parameters – one is guaranteed to never move in the wrong direction in parameter space. This is a highly desirable property for a numerical optimization algorithm.

While it is of course always possible to construct a lower bound to an objective function, nothing is gained if optimizing the lower bound is not significantly easier and/or faster to perform than optimizing the objective function directly. However, in the case of the Gaussian mixture model, it is possible to construct a lower bound for which the parameter vector maximizing it is given directly by analytic expressions. Therefore, the resulting algorithm effectively breaks up a difficult maximization problem (of the log likelihood function) into many smaller ones (of the lower bound) that are trivial to solve.

The trick exploited by the EM algorithm to construct its lower bound is based on the property of the logarithm that it is a concave function, that is, every chord connecting two points on its curve lies on or below that curve. Mathematically, this means that

$$\log[wx_1 + (1-w)x_2] \geq w\log x_1 + (1-w)\log x_2 \quad [18]$$

for any two points $x_1$ and $x_2$ and $0 \leq w \leq 1$. It is trivial to show that this also generalizes to more than two variables (the so-called Jensen's inequality):

$$\log\left(\sum_k w_k x_k\right) \geq \sum_k w_k \log x_k \quad [19]$$

where $w_k \geq 0$ and $\sum_k w_k = 1$, for any set of points $\{x_k\}$. This can now be used to construct a lower bound to the likelihood

Author's personal copy



function of the Gaussian mixture model as follows. Recalling that $p(\boldsymbol{d}|\boldsymbol{\theta}) = \prod_i \left[\sum_k \mathcal{N}\left(d_i|\mu_k, \sigma_k^2\right)\pi_k\right]$ (eqns [9] and [10]), we have that

$$\log p(\boldsymbol{d}|\boldsymbol{\theta}) = \log\left(\prod_i \left[\sum_k \mathcal{N}\left(d_i|\mu_k, \sigma_k^2\right)\pi_k\right]\right) \quad [20]$$

$$= \sum_i \log\left[\sum_k \mathcal{N}\left(d_i|\mu_k, \sigma_k^2\right)\pi_k\right] \quad [21]$$

$$= \sum_i \log\left[\sum_k \left(\frac{\mathcal{N}\left(d_i|\mu_k, \sigma_k^2\right)\pi_k}{w_k^i}\right) w_k^i\right] \quad [22]$$

$$\geq \underbrace{\sum_i \left[\sum_k w_k^i \log\left(\frac{\mathcal{N}\left(d_i|\mu_k, \sigma_k^2\right)\pi_k}{w_k^i}\right)\right]}_{Q(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}})} \quad [23]$$

for any set of weights $\{w_k^i\}$ that satisfy $w_k^i \geq 0$ and $\sum_k w_k^i = 1$ (the last step relies on eqn [19]). We now have a lower bound function $Q\left(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}}\right)$ that satisfies eqn [17], but not eqn [16], so we are not done yet. Instead of randomly assigning any valid $K$ weights $w_k^i$ to each voxel $i$ (one weight for each label $k$), we can satisfy eqn [16] by choosing the weights so that

$$w_k^i = \frac{\mathcal{N}\left(d_i|\tilde{\mu}_k, \tilde{\sigma}_k^2\right)\tilde{\pi}_k}{\sum_{k'} \mathcal{N}\left(d_i|\tilde{\mu}_{k'}, \tilde{\sigma}_{k'}^2\right)\tilde{\pi}_{k'}} \quad [24]$$

By filling these weights into the definition of our lower bound (eqn [23]), it is easy to check that eqn [16] is indeed fulfilled with this choice.

Setting the new model parameter estimate $\tilde{\boldsymbol{\theta}}$ to the parameter vector that maximizes the lower bound requires finding the location where

$$\frac{\partial Q\left(\boldsymbol{\theta}|\tilde{\boldsymbol{\theta}}\right)}{\partial \boldsymbol{\theta}} = 0,$$

which yields the following parameter update equations:

$$\begin{aligned} \tilde{\mu}_k &\leftarrow \frac{\sum_i w_k^i d_i}{\sum_i w_k^i} \\ \tilde{\sigma}_k^2 &\leftarrow \frac{\sum_i w_k^i \left(d_i - \tilde{\mu}_k\right)^2}{\sum_i w_k^i} \\ \tilde{\pi}_k &\leftarrow \frac{\sum_i w_k^i}{N} \end{aligned} \quad [25]$$

It is worth spending some time thinking about these equations. The EM algorithm searches for the ML parameters of the Gaussian mixture model simply by repeatedly applying the update rules of eqn [25], where the weights $w_k^i$ are defined in eqn [24]. These weights depend themselves on the current estimate of the model parameters, which explains why the algorithm involves iterating. By comparing eqn [24] to eqn [12], we see that the weights represent nothing but the posterior probability of the segmentation, given the current model parameter estimate. Thus, the EM algorithm repeatedly computes the type of probabilistic segmentation shown in Figure 2 based on its current parameter estimate and then updates the parameter estimate accordingly. The update rules of eqn [25] are intuitive: The mean and variance of the Gaussian distribution associated with the $k$th label are simply set to the weighted mean and variance of the intensities of those voxels currently attributed to that label; similarly the prior for each class is set to the fraction of voxels currently attributed to that class.

Figure 4 shows a few iterations of the EM algorithm searching for the ML parameters of the brain MR data shown in Figure 1(a).

## Modeling MR Bias Fields

Although the Gaussian mixture model is a very useful tool for tissue classification, it can often not be applied directly to MR images. This is because MR suffers from an imaging artifact that makes some image areas darker and other areas brighter than they should be. This spatially smooth variation of intensities is

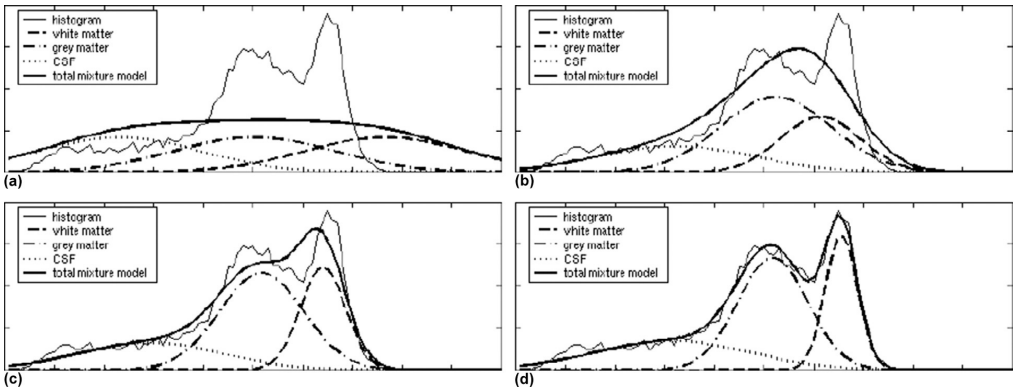



**Figure 4** Iterative improvement of the Gaussian mixture model parameters for the MR image of figure 1(a), using the EM algorithm: initialization (a) and parameter estimate after one (b), 10 (c) and 30 (d) iterations.

Author's personal copy



often referred to as MR 'intensity inhomogeneity' or 'bias field' and is caused by imaging equipment limitations and electrodynamic interactions with the object being scanned. The bias field artifact is dependent on the anatomy being imaged and its position in the scanner and is much more pronounced in the newest generation of scanners.

Since the Gaussian mixture model does not account for smoothly varying overall intensity levels within one and the same anatomical structure, it is very susceptible to segmentation errors when applied to typical MR data. However, this problem can be avoided by explicitly taking a model for the bias field artifact into account in the generative model. In particular, we can model the artifact as a linear combination of $M$ spatially smooth basis functions:

$$\sum_{m=1}^{M} c_m \phi_m^i \qquad [26]$$

where $\phi_m^i$ is shorthand for $\phi_m(\boldsymbol{x}_i)$, the value of the $m$th basis function evaluated at voxel $i$, which has spatial location $\boldsymbol{x}_i$. Suitable basis functions can be cosine functions, uniform B-spline basis functions, or something similar. We can then extend the Gaussian mixture model by still assigning each voxel an intensity drawn from a Gaussian distribution associated with its label, but further adding the bias model to the resulting intensity image to obtain the final bias field corrupted image $\boldsymbol{d}$ (because of the physics of MR, the bias field is better modeled as a multiplicative rather than an additive artifact. This can be taken into account by working with logarithmically transformed intensities in the models, instead of using directly the original MR intensities). With this model, we have

$$p(\boldsymbol{d}|\, \boldsymbol{l}, \boldsymbol{\theta}_d) = \prod_i \mathcal{N}\left(d_i - \sum_m c_m \phi_m^i \,|\mu_{l_i}, \sigma_{l_i}^2\right) \qquad [27]$$

with parameters $\boldsymbol{\theta}_d = (\mu_1, \ldots, \mu_K, \sigma_1^2, \ldots, \sigma_K^2, c_1, \ldots, c_M)^{\mathrm{T}}$, which consist not only of the parameters associated with the Gaussian distributions but also additionally of the $M$ coefficients of the bias field basis functions, $c_m$.

As was the case with the original Gaussian mixture model, model parameter estimation can be performed conveniently by iteratively constructing a lower bound to the log likelihood function. Following the exact same procedure as in the previous section, it can be shown (Van Leemput et al., 1999a; Wells et al., 1996) that constructing the lower bound involves computing the following weights (note the dependency on the bias field parameters in this case):

$$w_k^i = \frac{\mathcal{N}\left(d_i - \sum_m \tilde{c}_m \phi_m^i \,|\tilde{\mu}_k, \tilde{\sigma}_k^2\right)\tilde{\pi}_k}{\sum_{k'} \mathcal{N}\left(d_i - \sum_m \tilde{c}_m \phi_m^i \,|\tilde{\mu}_{k'}, \tilde{\sigma}_{k'}^2\right)\tilde{\pi}_{k'}} \qquad [28]$$

Subsequently maximizing the lower bound is more complicated than in the Gaussian mixture model, however, because setting the derivative with respect to the parameter vector $\boldsymbol{\theta}$ to zero no longer yields analytic expressions for the parameter update rules. If we keep the bias field parameters fixed at their current values $\tilde{c}_m$, and only maximize the lower bound with respect to the Gaussian mixture model parameters, we obtain

$$\begin{aligned} \tilde{\mu}_k &\leftarrow \frac{\sum_i w_k^i \left(d_i - \sum_m \tilde{c}_m \phi_m^i\right)}{\sum_i w_k^i} \\ \tilde{\sigma}_k^2 &\leftarrow \frac{\sum_i w_k^i \left(d_i - \sum_m \tilde{c}_m \phi_m^i - \tilde{\mu}_k\right)^2}{\sum_i w_k^i} \\ \tilde{\pi}_k &\leftarrow \frac{\sum_i w_k^i}{N} \end{aligned} \qquad [29]$$

Similarly, keeping the Gaussian mixture model parameters fixed at their current values, the bias field parameters maximizing the lower bound are given by

$$\tilde{\boldsymbol{c}} \leftarrow \left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{S} \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{S} \boldsymbol{r} \qquad [30]$$

where

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_1^1 & \phi_2^1 & \cdots & \phi_M^1 \\ \phi_1^2 & \phi_2^2 & \cdots & \phi_M^2 \\ \vdots & \vdots & \cdot & \vdots \\ \phi_1^N & \phi_2^N & \cdots & \phi_M^N \end{pmatrix} \qquad [31]$$

and

$$\begin{aligned} s_k^i &= \frac{w_k^i}{\tilde{\sigma}_k^2}, \quad s_i = \sum_k s_k^i, \quad \boldsymbol{S} = \mathrm{diag}(s_i), \quad \tilde{d}_i = \frac{\sum_k s_k^i \tilde{\mu}_k}{\sum_k s_k^i}, \\ \boldsymbol{r} &= \begin{pmatrix} d_1 - \tilde{d}_1 \\ \vdots \\ d_N - \tilde{d}_N \end{pmatrix} \end{aligned} \qquad [32]$$

Since eqns [29] and [30] depend on one another, one could in principle try to maximize the lower bound by cycling through these two equations, one at a time, until some convergence criterion is met. However, the desirable property of the EM algorithm to never decrease the value of the likelihood function with each new iteration still holds even when the lower bound is not maximized but merely improved. Therefore, a more efficient strategy is to construct the lower bound by computing the weights $w_k^i$ (eqn [28]) and then updating the Gaussian mixture model parameters (eqn [29]) and subsequently the bias field parameters (eqn [30]) only once to merely improve it. After that, a new lower bound is constructed by recomputing the weights, which is again improved by updating the model parameters, etc., until convergence. Such an optimization strategy of only partially optimizing the EM lower bound is known as so-called generalized EM.

The interpretation of the update equations is again very intuitive (Van Leemput et al., 1999a; Wells et al., 1996), but outside the scope of this article. Suffice it to say that by extending the Gaussian mixture model with an explicit model for the bias field artifact this way, it is possible to obtain high-quality segmentations of MR scans without errors caused by intensity inhomogeneities, as illustrated in Figure 5.

## Further Model Extensions

Although we have only described tissue classification techniques for unicontrast data so far (i.e., a single scalar intensity value for each voxel), the generative models can easily be extended to also handle multicontrast MR scans. In that scenario, the univariate Gaussian distributions are simply replaced with their multivariate equivalents. Furthermore, rather than using a single Gaussian to represent the intensity distribution of

Author's personal copy



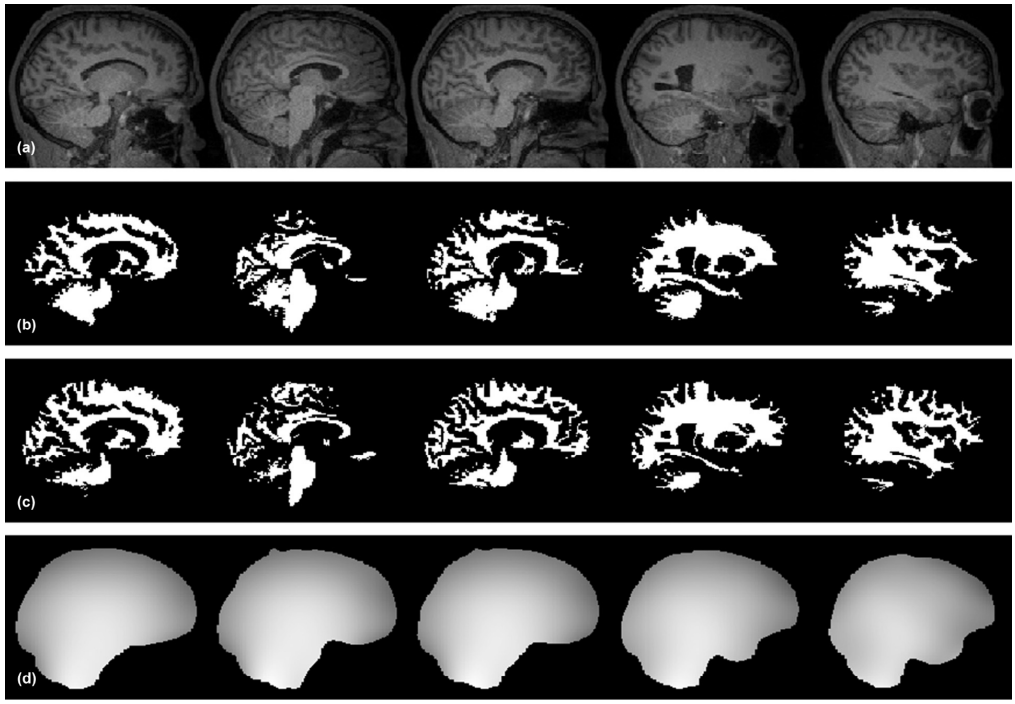



**Figure 5** Explicit modeling and estimating the bias field artifact in MR scans often improves segmentation results considerably. Shown are a few sagittal slices from a brain MR scan (a); the posterior probability for white matter using the standard Gaussian mixture model (b); the same when a bias field model is explicitly taken into account (c); and the automatically estimated bias field model (d). Note the marked improvement in segmentation accuracy in the upper parts of the brain.

any given label, a mixture of two or three Gaussians can provide more realistic intensity distribution models and yield more accurate segmentation results (Ashburner & Friston, 2005; Puonti, Iglesias, & Van Leemput, 2013).

Another class of extensions to the generative models covered in this article concentrates on the employed spatial model, that is, the segmentation prior $p(\boldsymbol{l}|\boldsymbol{\theta}_l)$. As a result of the rather simplistic modeling assumptions of the prior used so far (eqn [5]), a voxel's posterior probability of belonging to each of the $K$ structures is computed using only the local intensity of the voxel itself (eqn [12]). Although this works quite well in some applications, there is often an intensity overlap between the to-be-segmented structures, causing segmentation errors in such a purely intensity-driven strategy.

One possible improvement to $p(\boldsymbol{l}|\boldsymbol{\theta}_l)$ is the so-called Markov random field prior, which in typical usage encourages the different labels to occur in spatial clusters, rather than being scattered randomly throughout the image area (Held et al., 1997; Marroquin et al., 2002; Van Leemput et al., 1999b; Zhang et al., 2001). Although these priors have some attractive computational properties, they do not encode any information about the shape, organization, and spatial relationships of real neuroanatomical structures.

More powerful models can be obtained through so-called probabilistic atlases – either as stand-alone models or in combination with Markov random field priors – which encode prior anatomical knowledge of where to expect each of the tissue types in a typical human brain. Such atlases are constructed by spatially coregistering a large number of manually annotated brain scans and counting the frequencies of occurrence of the different tissue types. The resulting atlas is then brought into spatial correspondence with an image to be segmented, either as a preprocessing step (Van Leemput et al., 1999a) or as part of the model parameter estimation process within the generative modeling framework (Ashburner & Friston, 2005; Fischl, Salat et al., 2004; Pohl et al., 2006; Puonti et al., 2013). Either way, the frequencies are reformatted to obtain spatially varying prior probabilities $\pi_k^i$ for every class $k$ in every voxel $i$, as shown in **Figure 6**. These prior probabilities $\pi_k^i$ are then used in place of the generic $\pi_k$ in every equation of the segmentation models of this article, yielding voxel classifications that no longer depend solely on the voxels' local intensity alone but also on their spatial location. Furthermore, the priors $\pi_k^i$ unambiguously associate segmentation classes to predefined anatomical structures and can be used to automatically initialize the iterative update equations of the EM optimizers, even in multicontrast data where initialization is otherwise difficult. Finally, the spatial priors are also typically used to discard voxels that are of no interest, such as the muscle, skin, or fat in brain MR scans. As a result, the use of the spatial priors

Author's personal copy



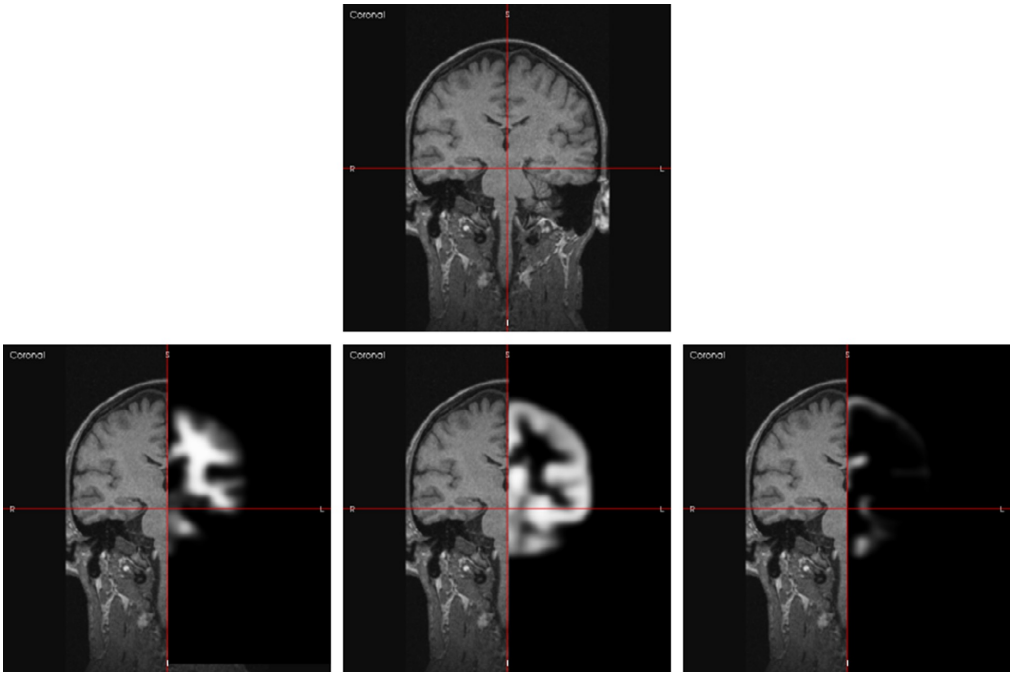



**Figure 6** Illustration of a probabilistic atlas aligned with an image-to-be-segmented. Top: anatomical scan to be segmented. Bottom: spatially varying prior probability maps of white matter, gray matter, and cerebrospinal fluid, overlaid on the anatomical scan for illustration purposes. Bright and dark intensities correspond to high and low probabilities, respectively.

$\pi_k^i$ contributes greatly to the overall robustness and practical value of the tissue classification models discussed in this article.

*See also:* **INTRODUCTION TO METHODS AND MODELING**: Intensity Nonuniformity Correction.

## References


Anbeek, P., Vincken, K. L., van Bochove, G. S., van Osch, M. J. P., & van der Grond, J. (2005). Probabilistic segmentation of brain tissue in MR imaging. *NeuroImage*, *27*(4), 795–804.

Ashburner, J., & Friston, K. J. (1997). Multimodal image coregistration and partitioning – A unified framework. *NeuroImage*, *6*(3), 209–217.

Ashburner, J., & Friston, K. J. (2005). Unified segmentation. *NeuroImage*, *26*, 839–885.

Awate, S. P., Tasdizen, T., Foster, N., & Whitaker, R. T. (2006). Adaptive Markov modeling for mutual-information-based, unsupervised MRI brain-tissue classification. *Medical Image Analysis*, *100*(5), 726–739.

Fischl, B., Salat, D. H., Busa, E., Albert, M., Dieterich, M., Haselgrove, C., et al. (2002). Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, *33*, 341–355.

Fischl, B., Salat, D. H., van der Kouwe, A. J. W., Makris, N., Ségonne, F., Quinn, B. T., et al. (2004). Sequence-independent segmentation of magnetic resonance images. *NeuroImage*, *23*, S69–S84.

Fischl, B., van der Kouwe, A., Destrieux, C., Halgren, E., Segonne, F., Salat, D. H., et al. (2004). Automatically parcellating the human cerebral cortex. *Cerebral Cortex*, *140*(1), 11–22.

Greenspan, H., Ruf, A., & Goldberger, J. (2006). Constrained gaussian mixture model framework for automatic segmentation of MR brain images. *IEEE Transactions on Medical Imaging*, *250*(9), 1233–1245.

Guillemaud, R., & Brady, M. (1997). Estimating the bias field of MR images. *IEEE Transactions on Medical Imaging*, *160*(3), 238–251.

Held, K., Kops, E. R., Krause, B. J., Wells, W. M., III, Kikinis, R., & Muller-Gartner, H. W. (1997). Markov random field segmentation of brain MR images. *IEEE Transactions on Medical Imaging*, *160*(6), 878–886.

Hunter, D. R., & Lange, K. (2004). A tutorial on MM algorithms. *The American Statistician*, *580*(1), 30–37.

Lorenzo-Valdés, M., Sanchez-Ortiz, G. I., Elkington, A. G., Mohiaddin, R. H., & Rueckert, D. (2004). Segmentation of 4D cardiac MR images using a probabilistic atlas and the EM algorithm. *Medical Image Analysis*, *8*(3), 255–265.

Marroquin, J. L., Vemuri, B. C., Botello, S., Calderon, F., & Fernandez-Bouzas, A. (2002). An accurate and efficient Bayesian method for automatic segmentation of brain MRI. *IEEE Transactions on Medical Imaging*, *210*(8), 934–945.

Menze, B., Van Leemput, K., Lashkari, D., Weber, M. A., Ayache, N., & Golland, P. (2010). A generative model for brain tumor segmentation in multi-modal images. *Medical Image Computing and Computer-Assisted Intervention-MICCAI*, *2010*(6362), 151–159.

Pham, D. L., & Prince, J. L. (1999). Adaptive fuzzy segmentation of magnetic resonance images. *IEEE Transactions on Medical Imaging*, *18*, 737–752.

Pohl, K. M., Bouix, S., Nakamura, M., Rohlfing, T., McCarley, R. W., Kikinis, R., et al. (2007). A hierarchical algorithm for MR brain image parcellation. *IEEE Transactions on Medical Imaging*, *260*(9), 1201–1212.

Pohl, K. M., Fisher, J., Grimson, E. L., Kikinis, R., & Wells, W. M. (2006). A Bayesian model for joint segmentation and registration. *NeuroImage*, *31*, 228–239.

Prastawa, M., Bullitt, E., Ho, S., & Gerig, G. (2004). A brain tumor segmentation framework based on outlier detection. *Medical Image Analysis*, *8*, 275–283.

Puonti, O., Iglesias, J. E., & Van Leemput, K. (2013). Fast, Sequence Adaptive Parcellation of Brain MR Using Parametric Models. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013* (pp. 727–734). Berlin/ Heidelberg: Springer.

Author's personal copy



Rajapakse, J. C., Giedd, J. N., & Rapoport, J. L. (1997). Statistical approach to segmentation of single-channel cerebral MR images. *IEEE Transactions on Medical Imaging*, *16*, 176–186.

Sabuncu, M. R., Yeo, B. T. T., Van Leemput, K., Fischl, B., & Golland, P. (2010). A generative model for image segmentation based on label fusion. *IEEE Transactions on Medical Imaging*, *290*(10), 1714–1729.

Van Leemput, K., Maes, F., Vandermeulen, D., Colchester, A., & Suetens, P. (2001). Automated segmentation of multiple sclerosis lesions by model outlier detection. *IEEE Transactions on Medical Imaging*, *200*(8), 677–688.

Van Leemput, K., Maes, F., Vandermeulen, D., & Suetens, P. (1999a). Automated model-based bias field correction of MR images of the brain. *IEEE Transactions on Medical Imaging*, *180*(10), 885–896.

Van Leemput, K., Maes, F., Vandermeulen, D., & Suetens, P. (1999b). Automated model-based tissue classification of MR images of the brain. *IEEE Transactions on Medical Imaging*, *180*(10), 897–908.

Warfield, S. K., Kaus, M., Jolesz, F. A., & Kikinis, R. (2000). Adaptive, template moderated, spatially varying statistical classification. *Medical Image Analysis*, *4*, 43–55.

Wells, W. M., III, Grimson, W. E.L, Kikinis, R., & Jolesz, F. A. (1996). Adaptive segmentation of MRI data. *IEEE Transactions on Medical Imaging*, *150*(4), 429–442.

Xue, H., Srinivasan, L., Jiang, S., Rutherford, M., Edwards, A. D., Rueckert, D., et al. (2007). Automatic segmentation and reconstruction of the cortex from neonatal MRI. *NeuroImage*, *380*(3), 461–477.

Zeng, X., Staib, L. H., Schultz, R. T., & Duncan, J. S. (1999). Segmentation and measurement of the cortex from 3D MR images using coupled surfaces propagation. *IEEE Transactions on Medical Imaging*, *180*(10), 927–937.

Zhang, Y., Brady, M., & Smith, S. (2001). Segmentation of brain MR images through a hidden Markov random field model and the expectation–maximization algorithm. *IEEE Transactions on Medical Imaging*, *20*, 45–57.

Appendix A

# Extension to multiple atlases

Provided we have access to multiple atlases built with the approach described in section 3.1.1, we can extend the modeling framework to support more than just a single probabilistic atlas. We assume that for each voxel $i$, the atlas index $m$ is drawn from $M$ possible atlases $m \in \{1, \ldots, M\}$ with equal probability, i.e., $p(m) = 1/M$. Once the atlas index is known, we draw the neuroanatomical label of the voxel from the appropriate atlas, i.e., $p_{i,m}(l_i|\mathbf{x}_m)$. As before, the likelihood model links each structure label to a mixture of Gaussians as in section 3.1.2.

The parameter optimization using the empirical Bayes approximation changes now slightly as we need to account for the multiple atlases. Specifically, the parameter posterior given the data is now written as:

$$\begin{aligned} p(\{\mathbf{x}_m\}, \boldsymbol{\theta}|\mathbf{D}) &\propto p(\mathbf{D}|\boldsymbol{\theta}, \{\mathbf{x}_m\})p(\{\mathbf{x}_m\})p(\boldsymbol{\theta}) \\ &= \left( \sum_{\mathbf{l},\mathbf{m}} p(\mathbf{D}, \mathbf{l}, \mathbf{m}|\boldsymbol{\theta}, \{\mathbf{x}_m\}) \right) p(\{\mathbf{x}_m\}) \\ &\propto \left( \prod_{i=1}^{I} \sum_{m=1}^{M} \sum_{k=1}^{K} p_i(\mathbf{d}_i|k, \boldsymbol{\theta}) p_{i,m}(k|\mathbf{x}_m) \right) \prod_{m=1}^{M} p(\mathbf{x}_m), \end{aligned}$$

where we have assumed a flat prior on the likelihoood parameters $\boldsymbol{\theta}$, and conditional independence between voxels given the atlas index and the label. Furthermore the deformation priors $p(\mathbf{x}_m)$ are assumed to be independent of each other.

The optimization problem now takes the form:

$$\begin{aligned}\{\{\hat{\mathbf{x}}_m\}, \hat{\boldsymbol{\theta}}\} &= \underset{\{\mathbf{x}_m\},\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathcal{L}(\{\mathbf{x}_m\}, \boldsymbol{\theta}) \\ \mathcal{L}(\{\mathbf{x}_m\}, \boldsymbol{\theta}) &= \left[\sum_{i=1}^{I} \log\left(\sum_{m=1}^{M}\sum_{k=1}^{K} p_i(\mathbf{d}_i|k,\boldsymbol{\theta}) p_{i,m}(k|\mathbf{x}_m)\right) + \sum_{m=1}^{M} \log p(\mathbf{x}_m)\right].\end{aligned} \tag{A.1}$$

Finding the ML parameter estimates proceeds in a similar fashion as in the single atlas case, by first keeping the mesh node positions $\{\mathbf{x}_m\}$ fixed and updating the likelihood model parameters $\boldsymbol{\theta}$, and subsequently fixing the likelihood model parameters and optimizing the positions of the mesh nodes. The gradient with respect to the node positions for atlas $m$ is now given by:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{x}_m} = -\beta \sum_{t=1}^{T} \frac{\partial U_t^{\kappa_m}(\mathbf{x}_m, \mathbf{x}_m^r)}{\partial \mathbf{x}_m} + \sum_{i=1}^{I} \frac{\sum_k p_i(\mathbf{d}_i|k,\boldsymbol{\theta}) \frac{\partial p_{i,m}(k|\mathbf{x}_m)}{\partial \mathbf{x}_m}}{\sum_{m,k} p_i(\mathbf{d}_i|k,\boldsymbol{\theta}) p_{i,m}(k|\mathbf{x}_m)}, \tag{A.2}$$

where $\kappa_m$ and $\mathbf{x}_m^r$ now refer to the connectivity and reference position of mesh $m$.

The GEM algorithm update equation are modified as follows, the expectation step becomes:

$$p_i(k^g, m|\mathbf{d}_i, \{\mathbf{x}_m\}, \boldsymbol{\theta}) = q_{i,m}^{k,g} = \frac{w_{k,g}\mathcal{N}\left(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i|\boldsymbol{\mu}_{k,g}, \boldsymbol{\Sigma}_{k,g}\right) p_{i,m}(k|\mathbf{x}_m)}{\sum_{m'=1}^{M}\sum_{k'=1}^{K} p_i(\mathbf{d}_i|k',\boldsymbol{\theta}) p_{i,m}(k'|\mathbf{x}_{m'})},$$

and for the maximization step we obtain:

$$\begin{aligned}
\boldsymbol{\mu}_{k,g} &\leftarrow \frac{\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k,g}(\mathbf{d}_i - \mathbf{C}\boldsymbol{\phi}^i)}{\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k,g}}, \quad w_{k,g} \leftarrow \frac{\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k,g}}{\sum_{g'=1}^{G_k}\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k',g}} \\
\boldsymbol{\Sigma}_{k,g} &\leftarrow \frac{\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k,g}(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)^T(\mathbf{d_i} - \boldsymbol{\mu}_{k,g} - \mathbf{C}\boldsymbol{\phi}^i)}{\sum_{i=1}^{I}\sum_{m=1}^{M} q_{i,m}^{k,g}} \\
\begin{pmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_N \end{pmatrix} &\leftarrow \begin{pmatrix} \mathbf{A}^T\mathbf{S}_{1,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{1,N} \\ \vdots & \ddots & \vdots \\ \mathbf{A}^T\mathbf{S}_{N,1}\mathbf{A} & \dots & \mathbf{A}^T\mathbf{S}_{N,N} \end{pmatrix}^{-1} \cdot \\
&\quad \begin{pmatrix} \mathbf{A}^T\left(\mathbf{S}_{1,1}\mathbf{r}_{1,1} + \dots + \mathbf{S}_{1,N}\mathbf{r}_{1,N}\right) \\ \vdots \\ \mathbf{A}^T\left(\mathbf{S}_{N,1}\mathbf{r}_{N,1} + \dots + \mathbf{S}_{N,N}\mathbf{r}_{N,N}\right) \end{pmatrix},
\end{aligned} \tag{A.3}$$

where as before the matrix $\mathbf{A}$ holds the basis functions estimated at each voxel. The diagonal matrix $\mathbf{S}_{h,j}$ for the input channel pair $(h, j)$ is now written as:

$$s_{i,m,k,g}^{h,j} = q_{i,m}^{k,g}\left(\boldsymbol{\Sigma}_{k,g}^{-1}\right)_{h,j}, \quad s_i^{h,j} = \sum_{k=1}^{K}\sum_{g=1}^{G_k}\sum_{m=1}^{M} s_{i,m,k,g}^{h,j}, \quad \mathbf{S}_{h,j} = \operatorname{diag}\left(s_i^{h,j}\right),$$

and the residue image $\mathbf{r}_{h,j} = \left(r_1^{h,j}, \dots, r_I^{h,j}\right)^T$ is defined as:

$$r_i^{h,j} = d_i^m - \frac{\sum_{k=1}^{K}\sum_{g=1}^{G_k}\sum_{m=1}^{M} s_{i,m,k,g}^{h,j}\left(\boldsymbol{\mu}_{k,g}\right)_j}{\sum_{k=1}^{K}\sum_{g=1}^{G_k}\sum_{m=1}^{M} s_{i,m,k,g}^{h,j}}.$$

Given the point estimates for the parameters the approximate MAP segmentation is obtained by finding the label which maximizes the approximate posterior for every voxel independently as:

$$\hat{l}_i = \underset{k}{\operatorname{argmax}} \sum_{g=1}^{G_k}\sum_{m=1}^{M} p_i(k^g, m|\mathbf{d}_i, \{\mathbf{x}_m\}, \boldsymbol{\theta}) \tag{A.4}$$

# Bibliography


[AAF00] J. Ashburner, R. L. J. Andersson, and J. K. Friston. Image registration using a symmetric prior–in three dimensions. *Human Brain Mapping*, 9(4):212–225, 2000.

[AEGG08] B. B. Avants, C. L. Epstein, M. Grossman, and J. 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.

[AF97] J. Ashburner and J. K. Friston. Multimodal image coregistration and partitioning – a unified framework. *NeuroImage*, 6(3):209–217, 1997.

[AF05] J. Ashburner and J. K. Friston. Unified segmentation. *NeuroImage*, 26(3):839–885, 2005.

[AHH+09] P. Aljabar, A. R. Heckemann, A. Hammers, V. J. Hajnal, and D. Rueckert. Multi-atlas based segmentation of brain images: Atlas selection and its effect on accuracy. *NeuroImage*, 46(3):726–738, 2009.

[AL13] J. A. Asman and A. B. Landman. Non-local statistical label fusion for multi-atlas segmentation. *Medical Image Analysis*, 17(2):194–208, 2013.

[AMoBOdS09] X. Artaechevarria, A. Munõz Barrutia, and C. 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.

[Ash01] J. Ashburner. *Computational neuroanatomy.* PhD thesis, University of London England, 2001.

[ATWF06] S. P. Awate, T. Tasdizen, R. T. Whitaker, and N. Foster. Adaptive Markov modeling for mutual-information-based, unsupervised MRI brain-tissue classification. *Medical Image Analysis*, 10(5):726–739, 2006.

[AVV08] P. Anbeek, K. L. Vincken, and M. A. Viergever. Automated MS-lesion segmentation by k- nearest neighbor classification. *The MIDAS Journal – MS lesion segmentation (MICCAI 2008 Workshop)*, 2008.

[BCA15] C. Blaiotta, M. J. Cardoso, and J. Ashburner. Variational inference for image segmentation. In *Bayesian and grAphical Models for Biomedical Imaging, Second International Workshop, BAMBI 2015.* 2015.

[BHC93] J. C. Bezdek, L. O. Hall, and L. P. Clarke. Review of MR image segmentation techniques using pattern recognition. *Medical Physics*, 20(4):1033–1048, 1993.

[BHP+04] R. L. Buckner, D. Head, J. Parker, A. Fotenos, D. Marcus, J. C. Morris, and A. Z. Snyder. A unified approach for morphometric and functional data analysis in young, old, and demented adults using automated atlas-based head size normalization: Reliability and validation against manual measurement of total intracranial volumes. *NeuroImage*, 23(2):724–738, 2004.

[BP08] P.-L. Bazin and D. L. Pham. Homeomorphic brain image segmentation with topological and statistical atlases. *Medical Image Analysis*, 12(5):616–625, 2008.

[CET98] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Computer Vision—ECCV'98*, pages 484–498. Springer Berlin Heidelberg, 1998.

[CH13] D. Chauveau and D. Hunter. ECM and MM algorithms for normal mixtures with constrained parameters. Technical report, 2013.

[CJFK89] V. S. Caviness Jr, P. A. Filipek, and D. N. Kennedy. Magnetic resonance technology in human brain science: blueprint for a program based upon morphometry. *Brain and Development*, 11(1):1–13, 1989.

[CMF+11] P. Coupé, V. J. Manjón, V. Fonov, J. Pruessner, M. Robles, and L. D. Collins. Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. *NeuroImage*, 54(3):940–954, 2011.

[DCB+10] G. Desjardins, A. C. Courville, Y. Bengio, P. Vincent, and O. Delalleau. Tempered Markov chain Monte Carlo for training of restricted Boltzmann machines. In *International Conference on Artificial Intelligence and Statistics*, pages 145–152, 2010.

[DHT+99] M. B. Dawant, L. S. Hartmann, P. J. Thirion, F. Maes, D. Vandermeulen, and P. Demaerel. Automatic 3-D segmentation of internal structures of the head in MR images using a combination of similarity and free-form transformations: Part I, methodology and validation on normal subjects. *IEEE Transactions on Medical Imaging*, 18(10):909–916, 1999.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal Of The Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[DMVS06] E. D'Agostino, F. Maes, D. Vandermeulen, and P. Suetens. A unified framework for atlas based brain image segmentation and registration. In *Biomedical Image Registration*, volume 4057, pages 136–143. 2006.

[DPGL+04] G. Dugas-Phocion, M.A. Gonzalez, C. Lebrun, S. Chanalet, C. Bensa, G. Malandain, and N. Ayache. Hierarchical segmentation of multiple sclerosis lesions in multi-sequence MRI. In *IEEE International Symposium on Biomedical Imaging: Nano to Macro*, pages 157–160 Vol. 1, 2004.

[EHWW13] S. M. A. Eslami, N. Heess, C. K. I. Williams, and J. Winn. The shape Boltzmann machine: a strong model of object shape. *International Journal of Computer Vision*, 107(2):155–176, 2013.

[FGG09] O. Freifeld, H. Greenspan, and J. Goldberger. Multiple sclerosis lesion detection using constrained GMM and curve evolution. *Journal of Biomedical Imaging*, pages 1–13, 2009.

[FI12] A. Fischer and C. Igel. An introduction to restricted Boltzmann machines. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 14–36. Springer Berlin Heidelberg, 2012.

[FRA+06] M. Filippi, M. A. Rocca, D. L. Arnold, R. Bakshi, F. Barkhof, N. De Stefano, F. Fazekas, E. Frohman, and J. S. Wolinsky.

EFNS guidelines on the use of neuroimaging in the management of multiple sclerosis. *European Journal of Neurology*, 13(4):313–325, 2006.

[FSB+02] B. Fischl, H. D. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. van der Kouwe, R. Killiany, D. Kennedy, S. Klaveness, A. Montillo, N. Makris, B. Rosen, and M. A. Dale. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33:341–355, January 2002.

[FSvdK+04] B. Fischl, D. H. Salat, A. J. W. van der Kouwe, N. Makris, F. Ségonne, B. T. Quinn, and A. M. Dale. Sequence-independent segmentation of magnetic resonance images. *Neuroimage*, 23:S69–S84, 2004.

[GCM+11] E. Geremia, O. Clatz, B. H. Menze, E. Konukoglu, A. Criminisi, and N. Ayache. Spatial Decision Forests for MS Lesion Segmentation in Multi-Channel Magnetic Resonance Images. *NeuroImage*, 57(2):378–90, July 2011.

[Gev06] T. Geva. Magnetic resonance imaging: historical perspective. *Journal of Cardiovascular Magnetic Resonance*, 8(4):573–580, 2006.

[GLFN+13] D. García-Lorenzo, S. Francis, S. Narayanan, D. L. Arnold, and D. L. Collins. Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging. *Medical Image Analysis*, 17(1):1 – 18, 2013.

[GLPA+11] D. García-Lorenzo, S. Prima, D. Arnold, D. L. Collins, and C. Barillot. Trimmed-likelihood estimation for focal lesions and tissue segmentation in multi-sequence MRI for multiple sclerosis. *IEEE Transactions on Medical Imaging*, 30(8):1455–1467, 2011.

[GMLB+14] T. Ge, N. Müller-Lenke, K. Bendfeldt, T. E. Nichols, and T. D. Johnson. Analysis of multiple sclerosis lesions via spatially varying coefficients. *The Annals of Applied Statistics*, 8:1095–1118, 2014.

[HF07] X. Han and B. Fischl. Atlas renormalization for improved brain MR image segmentation across scanner platforms. *IEEE Transactions on Medical Imaging*, 26(4):479–486, 2007.

[HHA+06] R. A. Heckemann, J. V. Hajnal, P. Aljabar, D. Rueckert, and A. Hammers. Automatic anatomical brain MRI segmentation

combining label propagation and decision fusion. *NeuroImage*, 33(1):115–126, 2006.

[Hin02] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[Hin12] G. E. Hinton. A practical guide to training restricted Boltzmann machines. In Grégoire Montavon, Genevieve B. Orr, and Klaus-Robert Müller, editors, *Neural Networks: Tricks of the Trade (2nd ed.)*, volume 7700 of *Lecture Notes in Computer Science*, pages 599–619. Springer, 2012.

[HKL+12] A. R. Heckemann, S. Keihaninejad, C. Ledig, P. Aljabar, D. Rueckert, V. J. Hajnal, and A. Hammers. Multi-atlas propagation with enhanced registration – maper. In *MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labeling*, pages 83–86, 2012.

[HLH+12] A. J. Holmes, P. H. Lee, M. O. Hollinshead, L. Bakst, J. L. Roffman, J. W. Smoller, and R. L. Buckner. Individual differences in amygdala-medial prefrontal anatomy link negative affect, impaired social functioning, and polygenic depression risk. *The Journal of Neuroscience*, 32(50):18087–18100, 2012.

[HOT06] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[IAN+15] J. E. Iglesias, J. C. Augustinack, K. Nguyen, C. M. Player, A. Player, M. Wright, N. Roy, M. P. Frosch, A. C. McKee, L. L. Wald, B. Fischl, and K. Van Leemput. A computational atlas of the hippocampal formation using ex vivo, ultra-high resolution MRI: Application to adaptive segmentation of in vivo MRI. *NeuroImage*, 115:117 – 137, 2015.

[IKZ+13] J. E. Iglesias, E. Konukoglu, D. Zikic, B. Glocker, K. Van Leemput, and B. Fischl. Is synthesizing MRI contrast useful for inter-modality analysis? In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, pages 631–638, 2013.

[ILTT11] J. E. Iglesias, C.-Y. Liu, P. M. Thompson, and Z. Tu. Robust brain extraction across datasets and comparison with publicly available methods. *IEEE Transactions on Medical Imaging*, 30(9):1617–1634, 2011.

[ISR+09] I. Isgum, M. Staring, A. Rutten, M. Prokop, M. A. Viergever, and B. van Ginneken. Multi-atlas-based segmentation with local decision fusion – application to cardiac and aortic segmentation in CT scans. *IEEE Transactions on Medical Imaging*, 28(7):1000–1010, 2009.

[ISV12] J. E. Iglesias, R. M. Sabuncu, and K. Van Leemput. A generative model for multi-atlas segmentation across modalities. In *9th IEEE International Symposium on Biomedical Imaging (ISBI), 2012*, pages 888–891, 2012.

[ISV13a] J. E. Iglesias, R. M. Sabuncu, and K. Van Leemput. Improved inference in Bayesian segmentation using Monte Carlo sampling: Application to hippocampal subfield volumetry. *Medical Image Analysis*, 17(7):766–778, 2013.

[ISV13b] J. E. Iglesias, R. M. Sabuncu, and K. Van Leemput. A unified framework for cross-modality multi-atlas segmentation of brain MRI. *Medical Image Analysis*, 17(8):1181–1191, 2013.

[JNS+13] B. C. Jones, G. Nair, C. D. Shea, C. M. Crainiceanu, I. C. M. Cortese, and D. S. Reich. Quantification of multiple-sclerosis-related brain atrophy in two heterogeneous MRI datasets using mixed-effects modeling. *NeuroImage: Clinical*, 3(0):171 – 179, 2013.

[KAA+09] A. Klein, J. Andersson, A. B. Ardekani, J. Ashburner, B. Avants, M.-C. Chiang, E. G. Christensen, L. D. Collins, J. Gee, P. Hellier, H. J. Song, M. Jenkinson, C. Lepage, D. Rueckert, P. Thompson, T. Vercauteren, P. R. Woods, J. J. Mann, and V. R. Parsey. Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. *NeuroImage*, 46(3):786–802, 2009.

[KBJ06] G. Kobelt, J. Berg, and B. Jönsson. Costs and quality of life in multiple sclerosis in Europe: method of assessment and analysis. *The European Journal of Health Economics*, 7(2):5–13, 2006.

[KGM+99] R. Kikinis, C. R.G. Guttmann, D. Metcalf, W. M. Wells, G. J. Ettinger, H. L. Weiner, and F. A. Jolesz. Quantitative follow-up of patients with multiple sclerosis using MRI: Technical aspects. *Journal of Magnetic Resonance Imaging*, 9(4):519–530, 1999.

[KRA+13] Z. Karimaghaloo, H. Rivaz, D. L. Arnold, D. L. Collins, and T. Arbel. Adaptive voxel, texture and temporal conditional random fields for detection of gad-enhancing multiple sclerosis lesions in brain MRI. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, volume 8151 of *Lecture*

*Notes in Computer Science*, pages 543–550. Springer Berlin Heidelberg, 2013.

[KSG98] A. Kelemen, G. Székely, and G. Gerig. Three-dimensional model-based segmentation of brain MRI. In *Workshop on Biomedical Image Analysis*, pages 4–13, 1998.

[LGRN09] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 609–616. ACM, 2009.

[LHA+12] C. Ledig, A. R. Heckemann, P. Aljabar, R. Wolz, V. J. Hajnal, A. Hammers, and D. Rueckert. Segmentation of MRI brain scans using MALP-EM. In *MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labeling*, pages 79–82, 2012.

[LIVL14] C. T. Larsen, J. E. Iglesias, and K. Van Leemput. N3 bias field correction explained as a bayesian modeling method. In *Bayesian and grAphical Models for Biomedical Imaging*, volume 8677 of *Lecture Notes in Computer Science*, pages 1–12. Springer International Publishing, 2014.

[LOC+12] X. Lladó, A. Oliver, M. Cabezas, J. Freixenet, J. C. Vilanova, A. Quiles, L. Valls, L. Ramió-Torrentà, and A. Rovira. Segmentation of multiple sclerosis lesions in brain MRI: A review of automated approaches. *Information Sciences*, 186(1):164 – 185, 2012.

[LW12] A. B. Landman and K. S. Warfield. Miccai 2012 workshop on multi-atlas labeling. In *15th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2012)*, pages 91–95, 2012.

[Min98] T. Minka. Expectation-maximization as lower bound maximization. *Tutorial published on the web at http://www-white. media. mit. edu/tpminka/papers/em. html*, 1998.

[MK10] R. Milo and E. Kahana. Multiple sclerosis: Geoepidemiology, genetics and the environment. *Autoimmunity Reviews*, 9(5):A387 – A394, 2010. Special Issue on The Environment Geoepidemiology and Autoimmune Diseases.

[MRV13] G. A. Milo, S. G. Rane, and K. F. Villa. The cost burden of multiple sclerosis in the United States: a systematic review of the literature. *Journal of Medical Economics*, 16(5):639 – 647, 2013.

[MTTT08] J. H. Morra, Z. Tu, A. W. Toga, and P. M. Thompson. Automatic segmentation of MS lesions using a contextual model for the MICCAI grand challenge. *The MIDAS Journal – MS lesion segmentation (MICCAI 2008 Workshop)*, 2008.

[NGS+09] M. Neema, Z. D. Guss, J. M. Stankiewicz, A. Arora, B. C. Healy, and R. Bakshi. Normal findings on brain fluid-attenuated inversion recovery MR images at 3T. *American Journal of Neuroradiology*, 30(5):911–916, 2009.

[NRM09] M. Norouzi, M. Ranjbar, and G. Mori. Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. In *Computer Vision and Pattern Recognition (CVPR), 2009.*, pages 2735–2742, 2009.

[NUZ00] L. G. Nyúl, J. K. Udupa, and X. Zhang. New variants of a method of MRI scale standardization. *IEEE Transactions on Medical Imaging*, 19(2):143–150, 2000.

[PBN+07] K. M. Pohl, S. Bouix, M. Nakamura, T. Rohlfing, R. W. McCarley, R. Kikinis, W. E. L. Grimson, M. E. Shenton, and W. M. Wells. A hierarchical algorithm for MR brain image parcellation. *IEEE Transactions on Medical Imaging*, 26(9):1201–1212, 2007.

[PCE+01] T. Paus, D. L. Collins, A. C. Evans, G. Leonard, B. Pike, and A. Zijdenbos. Maturation of white matter in the human brain: a review of magnetic resonance studies. *Brain Research Bulletin*, 54(3):255 – 266, 2001.

[PFG+06] M. K. Pohl, J. Fisher, L. E. W. Grimson, R. Kikinis, and M. W. Wells. A Bayesian model for joint segmentation and registration. *NeuroImage*, 31(1):228–239, 2006.

[PFS+03] M. S. Pizer, T. P. Fletcher, J. Sarang, A. Thall, Z. J. Chen, Y. Fridman, S. D. Fritsch, G. A. Gash, M. J. Glotzer, R. M. Jiroutek, C. Lu, E. K. Muller, G. Tracton, P. Yushkevich, and L. E. Chaney. Deformable M-reps for 3D medical image segmentation. *International Journal of Computer Vision*, 55(2–3):85–106, 2003.

[PG08] M. Prastawa and G. Gerig. Automatic MS lesion segmentation by outlier detection and information theoretic region partitioning. *The MIDAS Journal – MS lesion segmentation (MICCAI 2008 Workshop)*, 2008.

[PGLG05] M. Prastawa, G. Gerig, W. Lin, and J. H. Gilmore. Automatic segmentation of MR images of the developing newborn brain. *Medical Image Analysis*, 9(5):457–466, 2005.

[PHW+09] S. Pyne, X. Hu, K. Wang, E. Rossin, T.-I. Lin, L. M. Maier, C. Baecher-Allan, G. J. McLachlan, P. Tamayo, D. A. Hafler, P. L. De Jager, and J. P. Mesirov. Automated high-dimensional flow cytometric data analysis. *Proceedings of the National Academy of Sciences*, 106(21):8519–8524, 2009.

[PRG+05] E. Pagani, M. A. Rocca, A. Gallo, M. Rovaris, V. Martinelli, G. Comi, and M. Filippi. Regional brain atrophy evolves differently in patients with multiple sclerosis according to clinical phenotype. *American Journal of Neuroradiology*, 26(2):341–346, 2005.

[PSKJ11] B. Patenaude, M. S. Smith, N. D. Kennedy, and M. Jenkinson. A Bayesian model of shape and appearance for subcortical brain segmentation. *NeuroImage*, 56(3):907–922, 2011.

[RBMMJ04] T. Rohlfing, R. Brandt, R. Menzel, and R. C. 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.

[RCP13] S. Roy, A. Carass, and J. Prince. Magnetic resonance image example-based contrast synthesis. *IEEE Transactions on Medical Imaging*, 32(12):2348–2363, 2013.

[RHS11] F. Rousseau, A. P. Habas, and C. Studholme. A supervised patch-based approach for human brain labeling. *IEEE Transaction on Medical Imaging*, 30(10):1852–1862, 2011.

[RRMJ04] T. Rohlfing, B. D. Russakoff, and R. C. Maurer Jr. Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation. *IEEE Transactions on Medical Imaging*, 23(8):983–994, 2004.

[RSRF12a] M. Reuter, J. N. Schmansky, D. J. Rosas, and B. Fischl. Within-subject template estimation for unbiased longitudinal image analysis. *NeuroImage*, 61(4):1402–1418, 2012.

[RSRF12b] M. Reuter, N. J. Schmansky, H. D. Rosas, and B. Fischl. Within-subject template estimation for unbiased longitudinal image analysis. *Neuroimage*, 61(4):1402–1418, 2012.

[Sal08] R. Salakhutdinov. Learning and evaluating Boltzmann machines. Technical Report UTML TR 2008-002, Department of Computer Science, University of Toronto, June 2008.

[Sal15] R. Salakhutdinov. Learning deep generative models. *Annual Review of Statistics and Its Application*, 2(1):361–385, 2015.

[SBO+10] N. Shiee, P.-L. Bazin, A. Ozturk, D. S. Reich, P. A. Calabresi, and D. L. Pham. A topology-preserving approach to the segmentation of brain images with multiple sclerosis lesions. *NeuroImage*, 49(2):1524 – 1535, 2010.

[SH09] R. Salakhutdinov and G. Hinton. Deep Boltzmann machines. In *Artificial Intelligence and Statistics*, 2009.

[She94] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, 1994.

[SLAM08] J. Souplet, C. Lebrun, N. Ayache, and G. Malandain. An Automatic Segmentation of T2-FLAIR Multiple Sclerosis Lesions. In *The MIDAS Journal - MS Lesion Segmentation (MICCAI 2008 Workshop)*, 2008.

[SLC+08] M. Styner, J. Lee, B. Chin, M. Chin, O. Commowick, H. Tran, S. Markovic-Plese, V. Jewells, and S. Warfield. 3D Segmentation in the Clinic: A Grand Challenge II: MS lesion segmentation. In *The MIDAS Journal - MS Lesion Segmentation (MICCAI 2008 Workshop)*, 2008.

[Smo86] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. 1986.

[SYVL+10] M. R. Sabuncu, T. T. B. Yeo, K. Van Leemput, B. Fischl, and P. Golland. A generative model for image segmentation based on label fusion. *IEEE Transactions on Medical Imaging*, 29(10):1714–1729, 2010.

[SZE98] J.G. Sled, A.P. Zijdenbos, and A.C. Evans. A nonparametric method for automatic correction of intensity nonuniformity in MRI data. *IEEE Transactions on Medical Imaging*, 17(1):87–97, Feb 1998.

[TGCC14] V.-T. Ta, R. Giraud, D. L. Collins, and P. Coupé. Optimized patchmatch for near real time and accurate label fusion. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*, pages 105–112, 2014.

[Tie08] T. Tieleman. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1064–1071, 2008.

[TW15] X. Tomas-Fernandez and S. K. Warfield. A model of population and subject (MOPS) intensities with application to multiple sclerosis lesion segmentation. *IEEE Transactions on Medical Imaging*, 34(6):1349–1361, 2015.

[TWC+13] T. Tong, R. Wolz, P. Coupé, V. J. Hajnal, and D. Rueckert. Segmentation of MR images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. *NeuroImage*, 76(1):11–23, 2013.

[VL09] K. Van Leemput. Encoding probabilistic brain atlases using Bayesian inference. *IEEE Transactions on Medical Imaging*, 28(6):822–837, 2009.

[VLMVS03] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. A unifying framework for partial volume segmentation of brain MR images. *IEEE Transactions on Medical Imaging*, 22(1):105–119, 2003.

[VMVS99a] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated model-based bias field correction of MR images of the brain. *IEEE Transactions on Medical Imaging*, 18(10):897–908, 1999.

[VMVS99b] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated model-based tissue classification of MR images of the brain. *IEEE Transactions on Medical Imaging*, 18(10):885–896, 1999.

[VMVS01] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated segmentation of multiple sclerosis lesions by model outlier detection. *IEEE Transactions on Medical Imaging*, 20(8):677–688, 2001.

[VP15] K. Van Leemput and O. Puonti. *Tissue Classification*, volume 1, pages 373–381. Elsevier Inc, 2015.

[vRIA+10] M. E. van Rikxoort, I. Isgum, Y. Arzhaeva, M. Staring, S. Klein, A. M. Viergever, W. P. J. Pluim, and B. van Ginneken. Adaptive local multi-atlas segmentation: application to the heart and the caudate nucleus. *Medical Image Analysis*, 14(1):39–49, 2010.

[WHH08] M. Wels, M. Huber, and J. Hornegger. Fully automated segmentation of multiple sclerosis lesions in multispectral MRI. *Pattern Recognition and Image Analysis*, 18(2):347–350, 2008.

[WIG+96] M. W. Wells, III, L. E. W. Grimson, R. Kikinis, and A. F. Jolesz. Adaptive segmentation of MRI data. *IEEE Transactions on Medical Imaging*, 15(4):429–442, 1996.

[WRR13] N. Weiss, D. Rueckert, and A. Rao. Multiple Sclerosis Lesion Segmentation Using Dictionary Learning and Sparse Coding. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2013 - 16th International Conference, Nagoya, Japan, September 22-26, 2013, Proceedings, Part I*, pages 735–742, 2013.

[WSD+13] H. Wang, W. J. Suh, R. S. Das, J. Pluta, C. Craige, and A. P. Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):611–623, 2013.

[WWZ+13] G. Wu, Q. Wang, D. Zhang, F. Nie, H. Huang, and D. Shen. A generative probability model of joint label fusion for multi-atlas based brain segmentation. *Medical Image Analysis*, 2013.

[ZFE02] A. P. Zijdenbos, R. Forghani, and A. C. Evanc. Automatic "pipeline" analysis of 3-D MRI data for clinical trials: Application to multiple sclerosis. *IEEE Transactions on Medical Imaging*, 21(10):1280–1291, 2002.

[ZGC13] D. Zikic, B. Glocker, and A. Criminisi. Atlas encoding by randomized forests for efficient label propagation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, pages 66–73, 2013.

[ZGC14] D. Zikic, B. Glocker, and A. Criminisi. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. *Medical Image Analysis*, 18(8):1262–1273, 2014.