BORKOV: Automatic Segmentation of Multiple Sclerosis Lesions using K-Nearest Neighbors and Markov Random Field

A Research Project by

Franrey Anthony S. Saycon
Patricia Lorraine S. Sison

Submitted to the Department of Computer Science
College of Engineering
University of the Philippines

In Partial Fulfillment of the Requirements
for the Degree of
Bachelor of Science in Computer Science

College of Engineering
University of the Philippines
Diliman, Quezon City

May 2016

Table of Contents

List of Tables **iv**

List of Figures **v**

Abstract **vii**

Acknowledgement **viii**

1 Introduction **1**
1.1 Statement of the Problem 2
1.2 Significance of this Study 3
1.3 Scope of this Study 5

2 Theoretical Framework **6**
2.1 Automated MS-Lesion Segmentation by KNearest Neighbor Classification [9] 6
2.2 Efficient Interactive Brain Tumor Segmentation as Within-Brain kNN Classification [6] 7
2.3 Brain Surface Segmentation of Magnetic Resonance Images of the Fetus [5] 8
2.4 MS Lesion Segmentation using Markov Random Fields [7] 9

3 Methodology **11**
3.1 Datasets 11
3.2 KNN Module 11
3.2.1 Training 12
3.2.2 Testing 16
3.3 MRF Module 18

4 Experiments **21**
4.1 Environment 21
4.2 Specifications 22
4.3 Optimal Parameters 22
4.3.1 Features 22
4.3.2 Maximum A Posteriori (MAP) Iteration 30
4.3.3 Threshold 31
4.4 Experimental Results 32

5 Summary and Conclusion **38**

A Description of the Environment and Source Codes **40**
A.1 Environment 40
A.2 Interface Modules 40
A.2.1 Source Codes 40
A.2.2 Python Modules Used 43

Bibliography **45**

List of Tables

3.1 Parameters Used in MIPAV's BET Tool 13

4.1 Optimal Parameters 22
4.2 Performance of BORKOV in the 6 Datasets 34
4.3 Comparison of BORKOV to Other Segmentation Methods 36
4.4 Time Duration of BORKOV 37

List of Figures

1.1 Three Kinds of MRI Sequences 5

3.1 Brain mask depicted by the red outline 17
3.2 The Black Dot's Label: (Left) 100% Red, (Right) 62.5% Blue, 37.5% Red 18
3.3 7-pixel neighborhood (colored) containing the pixel of interest (yellow) 19

4.1 CHB Case 01 - Skullstripped, Normalized 24
4.2 CHB Case 02 - Skullstripped, Normalized 24
4.3 CHB Case 03 - Skullstripped, Normalized 25
4.4 UNC Case 01 - Skullstripped, Normalized 25
4.5 UNC Case 02 - Skullstripped, Normalized 26
4.6 UNC Case 03 - Skullstripped, Normalized 26
4.7 Scatter Plot of x, y, FLAIR 27
4.8 Scatter Plot of x, z, FLAIR 27
4.9 Scatter Plot of y, z, FLAIR 27
4.10 Scatter Plot of x, y, T1 27
4.11 KNN Sample Results among 4 Feature Combinations. Red to white are considered to be lesion regions of high probability. 28
4.12 The Different Trends of the Evaluation Methods: (a) MCC, (b) Accuracy, (c) Precision, (d) Miss Rate in 4 Features 30
4.13 Probabilistic Segmentation of 2 Training Scans with (a) 1, (b) 5, (c) 10 MAP Iterations 31

4.14 CHB Case 08 slice 250: (a) KNN Output, (b) After MRF Output, (c) KNN Compared with Manual Segmentation, (d) KNN-MRF Compared with Manual Segmentation 32
4.15 CHB Case 08 slice 260: (a) skullstripped, (b) probabilistic KNN segmentation, (c) labelled BORKOV segmentation, (d) probabilistic BORKOV segmentation, (e) manual segmentation, (f) comparison of BORKOV segmentation and manual segmentation 33

Abstract

Multiple sclerosis (MS) has been one of the devastating silent diseases of the nervous system. It is a chronic auto-immune disease in which the myelin and the nerve fibers are attacked and eventually damaged. Cure for MS has yet to be found and the key element to control the disease is detection. Magnetic resonance imaging (MRI), the preferred tool to establish a diagnosis of the disease, currently offers the most sensitive non-invasive way of imaging the brain. Through this particular tool, white matter (WM) lesions in the brain are seen, in which MS is characterized.

In monitoring the disease, manual segmentation of the brain MRI is conducted to visualize the progress of MS through the presence of WM lesions. Manual segmentation is considered to be time-consuming and labor-intensive, that is why automatic lesion segmentation is desirable. The problem remains on how those lesions will be divided and classified.

In this paper, we propose an automated method for segmentation of multiple sclerosis lesions. Our method uses a two-step approach: K-Nearest Neighbors (KNN) and Markov Random Field (MRF). Unlike its traditional process, we use KNN to produce a probabilistic result as the initial segmentation which is further improved by the second approach. In MRF, the local tissue information is used to conform to the biological fact that tissues tend to occur and agglomerate in locally contiguous patterns.

Acknowledgement

The researchers would like to thank their adviser, Dr. Prospero C. Naval Jr., for guiding them throughout the study.

Their laptops, MSI and Asus, deserve appreciation for having good specifications enough to run experiments of the study.

They would also like to thank Grayee, one of the resident cats in the Department of Computer Science building for trying to comfort them while they stress in making their thesis deliverables.

Chapter 1

Introduction

Multiple sclerosis (MS) is a disease in the central nervous system in which the immune system of the body attacks the protective sheath, called myelin, of the nerves either in the brain or in the spinal column. The myelin damage will eventually lead to the disruption of the communication between the brain and the body. This can lead to the loss of coordination with the limbs and can further lead to paralysis.

To help establish the diagnosis of MS, brain lesions must be detected. Since MS causes white matter (WM) inflammation, the WM lesions denote the manifestation of the disease. WM lesions are given more focus and emphasis in segmentation compared to the other brain tissue classifications, namely gray matter (GM) and cerebrospinal fluid (CSF). The use of magnetic resonance imaging (MRI) has been the preferred method in clinical practices to help establish the diagnosis of MS and to monitor the course of the disease. The brain MRI scans are used for qualitative and quantitative analyses.

Manuel segmentation of MS lesions is done for the assessment of the disease progression. However, this is considered challenging, time-consuming, and prone to observer bias. This problem gives rise to the need of an automatic segmentation of MS

lesions and thus, different approaches have been introduced by several researchers in the past. However, the performances of the past approaches have yet to meet the standard for clinical practice. We wish to address the issue by proposing an automatic segmentation method, called BORKOV, with a two-step approach: k-nearest neighbors (KNN) and Markov random field (MRF).

KNN is a lazy supervised learning algorithm making use of collected features and their assigned labels to create inferences to given sample test data. The process works by getting k minimum distances from a certain test data value to all the collected data values through a specific distance function. The label of the certain data is decided through getting the majority label of the k nearest distance features. To avoid ties, k is usually odd.

MRF is a graphical model for a joint probability distribution. Other than image segmentation, its application in vision is image restoration, image reconstruction, and edge detection. In classification of the brain tissues in MR images, MRF accommodates the following considerations: class labels are few and known in advance and brain MR images usually form coherent and continuous shapes.

1.1 Statement of the Problem

The goal of acquiring brain MRI of a patient is for diagnosing MS and/or for monitoring its progression. The quantitative analysis of the WM lesion load depends on the MRI sequence used. An MRI sequence is a number of radio-frequency pulses and gradients that result in a set of images with particular appearance. Thus, a particular MRI sequence that leads to the best quantitative analysis must be chosen.

The objective of automatic MS lesion segmentation is to label lesions and non-lesion tissues in brain MRI with little or no human intervention involved. We define segmentation as a composition of two simultaneous processes: division and classification. Division groups a set of random variables according to specific conditions, such as shared characteristics. Classification gives the meaningful labels to the groups.

The problem, therefore, lies within the ability of the machine to distinguish and trace WM lesions in a brain MRI through automatic segmentation. Like the observer in charge with manual segmentation, the machine must also first undergo a training phase in order familiarize itself with the features of a WM lesion before it segments a brain scan. This process consists of collecting feature vectors of lesions and non-lesions from a training set of brain MRI. Next, in the testing phase, the machine segments a brain MRI not included in the training set with respect to the known feature vectors. Finally, the initial segmentation is improved by having WM lesions agglomerate in locally contiguous patterns.

1.2 Significance of this Study

The Multiple Sclerosis Foundation estimates that about 2.5 million people in the world suffer from MS. No two people who have the yet-to-be-curable disease share the same symptoms. Diagnosis is important because a self-assessment of MS is more often than not unreliable and treatments must start as early as possible to slow the progression of MS. MRI plays a crucial role in diagnosing a patient with the disease and monitoring the progression.

For diagnosing MS, qualitative analysis is often done to visually assess the brain tissues shown in the brain MRI of a patient. Meanwhile, quantitative analysis is the

key element in the assessment of MS progression. In clinical trials, the WM lesions, the characterization of the disease, are manually segmented. This method is said to be labor-intensive and time-consuming because of the large number of MRI slices needed to extract the lesion load from. Moreover, manual segmentation is prone to observer bias. Intra-observer variability is the inconsistency of results produced by an observer who analyzed the same MRI scan at different times, while the inter-observer variability is the inconsistency of results produced by different observers who analyzed the same MRI scan.

Several studies have proposed their respective automatic segmentation methods to address these problems. However, the comparison results from these methods with manually segmented brain MR images (which serve as the ground truth) show less agreement compared to the comparison of results from independent observers with each other. Therefore, our study wishes to create BORKOV, an automatic segmentation method that is able to produce a result that shows more significant similarity with the manually segmented brain MR images, enough to emerge as a standard for clinical practice.

BORKOV produces a probabilistic segmentation of an MRI scan as opposed to labelled segmentations by most of the automatic MS lesion segmentation methods. A probabilistic segmentation shows a color map of an MR image, each pixel contains its lesion-likelihood probability. This kind of segmentation gives way to flexibility; it acquires a user specified confidence level percentage or threshold for the lesion segmentation.

1.3 Scope of this Study

There are different kinds of MRI sequences – T1-weighted, T2-weighted, and FLAIR, to name a few. The visual characteristics of the three are shown in figure 1.1. The FLAIR image possesses the same visual characteristics as of T2-weighted image, except the CSF appears dark. For our study, we used only T1 and FLAIR brain MR images. The reason for this choice is explained in Chapter 4 Experiments.

An MRI can be viewed in three different ways: axial, sagittal, and coronal. An axial plane transects the body from top to bottom, sagittal plane from side to side, and coronal plane from front to back. Only brain MR images transected with axial plane were used in this study since they are the most commonly used in MS lesion segmentation.

Our proposed segmentation method produces probabilistic and labelled segmentations; when a threshold is applied, BORKOV produces two classifications, namely, lesion and non-lesion. Healthy WM tissues, GM tissues, CSF, and air/background fall under the same category, non-lesion, since our study focuses solely on the segmentation of MS lesions.

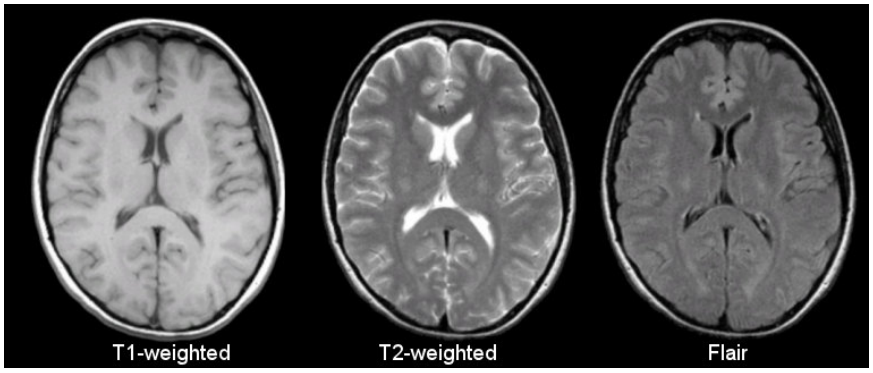

Figure 1.1: Three Kinds of MRI Sequences

Chapter 2

Theoretical Framework

2.1 Automated MS-Lesion Segmentation by KNearest Neighbor Classification [9]

Their research is primarily concerned with the use of solely KNN to automatically segment MS lesions. Their training and testing datasets were from the 2008 MICCAI MS Lesion Segmentation Challenge. This research suggests the use of FLAIR as the best sequence input for MS lesion segmentation. The choice of k is dependent upon the relation between the number of features and the number of cases. A small k causes the result being influenced by individual cases, while a large value of k makes the classification outcome smoother. In general, for the segmentation process, a large k is favorable. 5% of the total population of pixels in the different MR images belonging to the training dataset are used to be allocated in the memorized training set.

To have a binary segmentation, they made use of a threshold. This is simply the minimum probability value for a pixel to be considered a lesion. Their method discussed the problem of having a low precision (a metric that concerns with the true positive over the true positive and false negative) due to oversegmentation of lesions.

This could lead to the use of varying thresholds.

The problem with gold standards was discussed here. In the research, they had three gold standards to compare their ratings with. Despite the presence of two manual segmentations of different raters and the combined STAPLE segmentation of different automatic methods, it is still difficult to identify the most favorable segmentation. Brain abnormalities, such as MS lesions, have a large partial volume area, since their intensity changes gradually into normal tissue.

Therefore, the ultimate goal of the segmentation mostly determines the suitability of the segmentations. In large cohort studies, a structural over- or undersegmentation may not be problematic, as long as it is consistently performed. In this research, reproducibility is highly important, since a proper comparison between groups must be guaranteed. The proposed method is fully automated and reproducible.

2.2 Efficient Interactive Brain Tumor Segmentation as Within-Brain kNN Classification [6]

This research used KNN with MRF and conditional random field (CRF). This method shows that MRF and CRF are used to regularize the output of the KNN method. Their problem involves the automatic segmentation of brain tumor with the use of 6 features: spatial coordinates, and T1, T2 and FLAIR intensities. It is seen that spatial information can be a good feature in MRI segmentation problems from which we decide to take this into account.

KNN, although being the simplest algorithm, produced good results in this research and is relatively fast. In this research, the algorithm treats each brain as a separate dataset. This means it is immune to the multi-MRI disadvantages such

as the intensities of MR images are not standardized across MRI scanners. Also, manually segmented brain images for training are not so frequent and require lots of man power to build. Other obstacles include inter-slice intensity variations, differences in the noise produced by different MRI scanners, tumors-related problems when registering and aligning images, etc. In order to absolve this, they used several pre-processing method with high effort of tuning.

Unlike [9], their KNN implementation produces labelled segmentation, not probabilistic. This is the traditional KNN, where each pixel is decided through majority label of the k nearest datapoints. It was stated that segmentation accuracy can easily be improved by leveraging a model of the 3D spatial regularity of labels, which in this research suggests MRF and CRF.

2.3 Brain Surface Segmentation of Magnetic Resonance Images of the Fetus [5]

This study proposed a method for segmenting brain tissues of MR images of a fetus using Finite Gaussian Mixture Model (FGMM) and MRF. The aim of the algorithm is to map and classify multiple volumes (axial, coronal and sagittal acquisitions) into two main classes: brain tissue and non-brain tissue.

The FGMM approach classifies each voxel according to their intensities using 5 labels: two each for brain tissue and non-brain tissues classes, and one for transition voxels. The transition voxels represent voxels with uncertainty between GM and CSF. Voxels that fall under this category is re-classified into the two main classes (brain and non-brain) with the MRF scheme. The choice of using two classes for each tissue has been done empirically.

The MRF approach uses spatial information and no intensity information to redistribute and reclassify every voxel in the transition voxel label in the two tissues, brain and non-brain. Moreover, the intracranial cavity voxels wrongly classified as brain in the first approach, typically voxels not segmented by their skullstripping method, are corrected by MRF. Local tissue information is used, so a second order neighborhood system is exploited. The neighborhood is composed of 27 voxels: the voxel of interest, and the 26 voxels directly surrounding it (eight in the same slice, nine each in top and bottom slices). In order to find a new label of a voxel, energy must be minimized through Maximum a posteriori (MAP) of the probability of the voxel having a certain label.

The additional processes, namely surface reconstruction and extending MRF neighborhood, make use of the multiple volumes for a more improved segmentation of the fetal brain.

2.4 MS Lesion Segmentation using Markov Random Fields [7]

An MRF model is used to construct multispectral MRI (T1-weighted, T2-weighted and the proton density (PD) weighted image modalities) tissue intensities for the whole brain including the posterior fossa (PF) by incorporating voxel level spatial information in a standard anatomical space. Unlike other methods presented by previous researchers, all brain tissues are divided and segmented into their own classes instead of having lesions as outliers.

The objective is to classify each MRI voxel into one of the following classes: Background (Bk), White Matter (WM), Grey Matter (GM), Cerebrospinal Fluid (CSF),

T1-hypointense lesions ($T1_{les}$) and T2-hyperintense lesions ($T2_{les}$). Their MRF tissue classification approach is divided in into two parts: training and MRF proper.

In the training process, a three dimensional feature vector is used to denote the intensity of a voxel. The elements of the vector correspond to the voxel intensities in T1-weighted, T2-weighted and PD-weighted modalities. Intensity histograms of the brain tissues are approximated by a multivariate Gaussian distribution, respectively.

A brain MRI is modeled as an instance of an MRF and the goal is to obtain the best configuration of labels over all the voxels in the image. Using the local neighbourhood information to cluster voxels to their classes, a homogeneous and isotropic neighbourhood system is defined. The tissue class membership probability of a particular voxel depends on its six-voxel neighborhood: four from the same slice, one from the top slice, and one from the bottom slice, all of which has a Euclidean distance of one unit to the voxel of interest.

Chapter 3

Methodology

3.1 Datasets

A total of 20 MR image sets (each from a different patient) with manual segmentation (serves as the ground truth) were provided by 2 sources: 10 from Children's Hospital Boston (CHB) and 10 from University of North Carolina (UNC). The MR images from UNC were acquired on a Siemens 3T Allegra MRI scanner with slice thickness of 1 mm and in-plane resolution of 0.5 mm. No scanner information was provided about the CHB MR images. An MR image set of a patient consists of a T1-weighted image, a T2-weighted image, and a FLAIR image, all have 512x512x512 dimensions. To ease the segmentation process, all data has been rigidly registered to a common reference frame and resliced to isotrophic voxel spacing using B-spline based interpolation.

3.2 KNN Module

KNN is a lazy machine learning algorithm that makes use of memorized features for classifying data. Each unclassified data is compared to all memorized data using a specified distance metric. The label of a datapoint is then decided through collecting the majority label of the k nearest datapoints. In this research, the unlabelled data

points are the pixels in the MR image. We aim to label these pixels into two classes: lesion or non-lesion. Unlike traditional KNN, wherein the label is decided through a majority vote, we use a probabilistic output noting the likelihood of a certain pixel to be lesion pixel.

We used the Scikit-Learn module of Python to aid us in the implementation of our KNN algorithm. Scikit-Learn supports different types of algorithms that are used commonly in machine learning. Scikit-Learn provides different kinds of approach for KNN, and we chose to use the brute force implementation since tree structures implementation become counter-intuitive if used with high k values.

We modelled our training process in such a way that it utilized some disk space since incorporating several 512x512x512 MR images to the RAM is deemed impossible in our current machine. All our input training KNN data have special strings on top of the features taking note of the feature number, the means of the corresponding features, and the standard deviation of the corresponding features. Our training algorithm is divided into five parts: identification, segregation, retention, chunking, and standardization.

3.2.1 Training

70% of the datasets are used for training and 30% for testing. This means all 14 datasets (7 CHB and 7 UNC) were used in the training algorithm to create a memorized training set. Since it's counter-intuitive to use all the datapoints in each dataset, not to mention memory-heavy, we selected 5% of the data point population of each MR image only. All the selections are done randomly.

Identification

Identification is divided into two phases: skull stripping and memorization.

Skull Stripping Skull stripping is done using Medical Image Processing, Analysis, and Visualization (MIPAV) software. We used Brain Extraction Tool (BET) [12] for our skull stripping process. Through visual inspection of the resulting MR image, we obtained the optimal parameters (found in table 3.1). Non-brain tissue intensities are automatically set to 0, and this is accounted for in the memorization process. It is known that T1 is the best input for the BET [9], so the T1 images from the 14 datasets are used for memorization.

Parameter	Value
Iterations	1000
Depth	7
Image Influence	0.1
Stiffness	0.5

Table 3.1: Parameters Used in MIPAV's BET Tool

Memorization Given that the raw MR image inputs are already skullstripped, we now aim to save all the coordinates of interest. This is simply the coordinates of the pixels that are part of the brain. This is easily done because of BET. We call this data, the memory. Each MR image has their own corresponding memory.

Segregation

We need to know whether the coordinates of each content as a lesion or non-lesion. This is necessary since the memorized data in KNN should have their corresponding labels. The memories of each MR image are then divided into two: lesion memory

and non-lesion memory using the corresponding ground truths of the MR Image. There are two raters found in the datasets: the UNC rater and CHB rater. We used the CHB rater because of inter-rater differences and the CHB rater, upon visual inspection, found more lesions than UNC. Also, only the CHB rater was able to manually segment all datasets, unlike UNC that only segmented the UNC datasets. This is to have one general rater for our datasets.

There's also the problem of label bias. It is a fact that there are more non-lesion pixels than lesion pixels. The KNN classifier, if this is the case, becomes biased towards non-lesion pixels because of their vast majority. Part of the segregation process is to randomly choose n non-lesion pixels to be saved in the non-lesion memory, where n is the population of the corresponding lesion-memory. In effect, the two memories have the same population.

Retention

It was discussed that saving the whole population is counter-intuitive. So we chose only 5% of the population in each of the memories and thus, resulting to the 5% of the total population through summation. Let's define these memory samples as chunks. As a result, each corresponding lesion-memory and non-lesion memory have their own corresponding chunks. The chunks are used for the next process.

Chunking

This is the process of combining all the chunks to create the final memorized training set. Before combination, the coordinates in each chunk is processed and undergone feature extraction. The feature to be used is discussed in Chapter 4 Experiments. As a result, we now have a final memorized training set that has the features extracted

with their corresponding label.

Standardization

This is the final process of the training algorithm. This process is the standardization of the final memorized training set. Using the Python module NumPy, we can easily get the mean and standard deviation of a given data range which are the prerequisites for standardization. Standardization means transforming the data into their corresponding z-scores. The whole corresponding feature range now has its mean to 0 and a variance scaling of 1. This is necessary when there are a combination of features since it solves the problem of a feature having a greater influence in the distance function. This enables the features to contribute equally. The standardization formula is shown in 3.2.1:

$$z = \frac{x - \mu}{\sigma} \tag{3.2.1}$$

Each feature range has their own mean and standard deviation. Their corresponding means and standard deviations were used for getting their z-scores. The spatial information however was standardized in a little different way. Instead of having the mean and standard deviation in the current memorized training set, we used the fact that each training dataset is 512x512x512 in dimension.

KNN is heavily reliant on the initial memorized training data fed to it, so having unbalanced features often lead to unreliable results. One more importance of generalization of data is because MR images, in reality, are not a product of universal procedures. There are differences in the machine used, dimensionality, slice thickness, intensities, etc.

3.2.2 Testing

By now, we have created the training data to be fed to the classifier. The k input is user-specified. The training data was parsed having taken note of the number of features, corresponding mean and standard deviation of the features. The optimal k is discussed later.

We used the 30% of the datasets for testing – 3 CHB datasets and 3 UNC datasets – with the manual segmentation of the CHB rater as ground truth. The test inputs are also assumed to be skullstripped using the same method of BET from MIPAV with the same parameters. The necessity of skullstripping is discussed in the Brain Mask Creation. We aim to have a probability array of the same dimensions of the MR image. The probability is the likelihood of a certain pixel to be a lesion.

To optimize speed, since each pixel is independently processed, with the help of Scikit-Learn, BORKOV was set to parallel program, which the number of parallel processes is dependent on the number of cores the machine has. The testing process is divided into two: Brain Mask Creation and the KNN-Proper.

Brain Mask Creation

Given that most pixels in a brain MRI are just background pixels, brain masking is done to reduce the pixels processed in BORKOV. The brain mask is simply a collection of rectangle coordinates with each corresponding to a slice number.

Given a slice S, the minimum and maximum of x coordinates and of y coordinates containing brain tissues are identified and used for the four coordinates of the brain mask of slice S. That is, a minimum spatial coordinate denotes the position of the first discovered brain tissue from the edge of S, or the first discovered pixel with a

non-zero intensity (background has a zero intensity). A maximum spatial coordinate, meanwhile, denotes the position of the last discovered brain tissue.

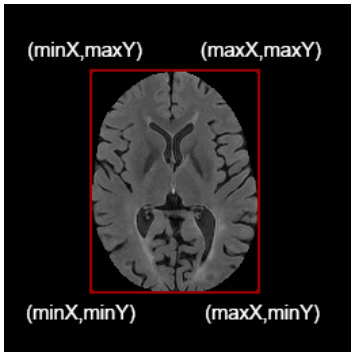

Figure 3.1: Brain mask depicted by the red outline

BORKOV ignores not only the areas outside the brain mask of a slice, but also all slices that contain no brain tissues. Pixels not covered by the brain mask automatically receive a 0% lesion-likelihood probability. This process reduces the processing time of the segmentation significantly.

KNN Proper

We now have a KNN classifier and a brain mask that aids this process. The process loops over all the pixels bounded by the brain mask and have their features extracted. There's also a need to standardize the features extracted from the test pixels. We used the mean and standard deviation vectors from the memorized training set to standardize the incoming data. This generalizes the test features to fit into our memorized training set.

The distance measure used was Euclidean distance simply because of the observations in our experiments when it comes to label separation. The Euclidean distance formula is shown in 3.2.2:

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3.2.2}$$

As discussed earlier, our output is the lesion likelihood probability of the pixel. This is described by formula 3.2.3, where n_l is the number of lesion neighbors and k is the specified k.

$$p = \frac{n_l}{k} \tag{3.2.3}$$

Figure 3.2 shows the summary of the KNN algorithm.

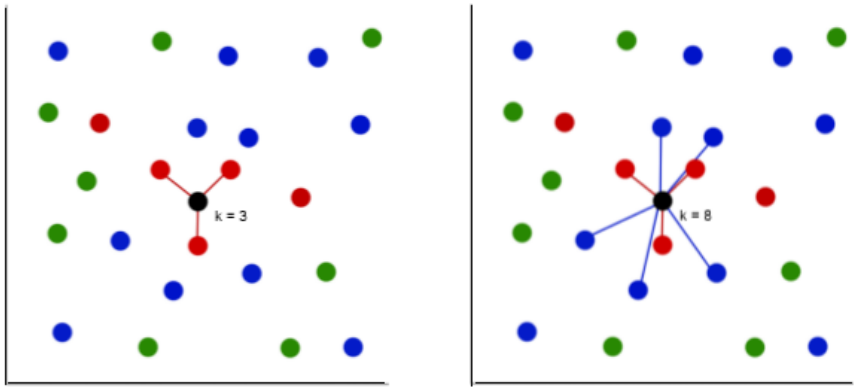

Figure 3.2: The Black Dot's Label: (Left) 100% Red, (Right) 62.5% Blue, 37.5% Red

3.3 MRF Module

A Markov random field is a graphical model of a joint probability distribution. It consists of an undirected graph in which the nodes represent random variables. An MRF satisfies any of the following properties: (1) any two non-adjacent variables are conditionally independent given all other variables (pairwise Markov property), (2) a variable is conditionally independent of all other variables given its neighbors (local

Markov property) and (3) any two subsets of variables are conditionally independent given a separating subset (global Markov property). For this particular image segmentation, the local Markov property is used to abide with the tissue behavior of occurring and agglomerating in locally contiguous patterns.

The probabilistic output from KNN is assumed to be a realization of several MRFs. Using this initial result, wherein a pixel i contains the probability of being a lesion $x_i = L$, MRF finds the optimal probability of a pixel through local Markov property. The probability is depicted by the following equation:

$$P(x_i|x_{S-i}) = P(x_i|x_{N_i}) \tag{3.3.1}$$

where an area in an MRI scan S is related with a neighborhood system $N = \{N_i, i \in S\}$, N_i is the set of sites neighboring i, with $i \notin N_i$ and $i \in N_j \iff j \in N_i$. A neighborhood of i consists of six neighbors: four in-slice neighbours and the two neighbours from preceding and succeeding slices. The Euclidean distances of i to each of its neighbors are all 1 unit.

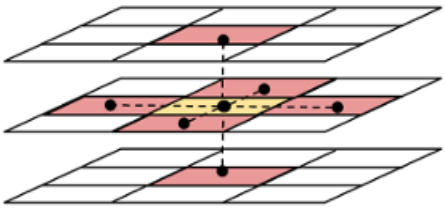

Figure 3.3: 7-pixel neighborhood (colored) containing the pixel of interest (yellow)

By Hammersley-Clifford theorem, the random field is described by a Gibbs distribution:

$$P(x_i|x_{N_i}) = \frac{e^{-U(x_i)}}{\sum_{x \in X} e^{-U(x_i)}} \tag{3.3.2}$$

where $U(x_i)$ is the energy function. A new label map is found through iterations of

Maximum a posteriori (MAP) of equation 3.3.2 by energy minimization:

$$\hat{x_i} = \underset{0 \leq P(x_i = L) \leq 1}{\operatorname{argmin}} (U(x_i)) \tag{3.3.3}$$

The energy function is defined as:

$$U(x_i) = \sum_{\forall j \in N_i} V_{ij}(x_i, x_j) \tag{3.3.4}$$

where V_{ij} is the affinity function, which shows how much two neighboring pixels i and j agree. It is defined by the Bhattacharyya distance that measures the similarity of the two discrete probability distributions:

$$V_{ij}(x_i, x_j) = -ln(\sum_{\forall x \in T} \sqrt{P(x_i)P(x_j)}) \tag{3.3.5}$$

where set $T = \{L, NL\}$, L is the label for lesions, NL for non-lesions.

To obtain the actual label x_i for a probabilistic output, a threshold is applied. The binary segmentation works in such a way that when $P(x_i = L)$ is greater than or equal to the threshold value, $x_i = L$; otherwise, $x_i = NL$.

Chapter 4

Experiments

4.1 Environment

BORKOV is platform independent. It is run in the command prompt or terminal. The environment under which the simulations for this paper are executed is specified below:

- Hardware: Intel Core i7 with 8 cores, at least 8GB RAM (at least 16GB RAM needed for skullstripping)
- Software: Windows 10 Operating System
- Language: Python 2.7.11 (64-bit)
- System: SimpleITK
- Tools: scikit-learn, matplotlib, NumPy, SciPy, TkInter
- Modules: __future__, datetime, gc, math, multiprocessing, shutil, os, sys
- User Interface: MIPAV - Medical Image Processing, Analysis and Visualization (for skullstripping and viewing medical images)

4.2 Specifications

BORKOV is trained to automatically segment MS lesions using as input a skull-stripped FLAIR MRI from scanners of CHB or UNC. The file extension of the input must be supported by SimpleITK. The program also requires a specific memorized training set file, namely Spatial_FLAIR_5p_Standardized.csv.

Using its flexibility, the program is set to accept user-specified parameters:

- K: the number of minimum distances from a datapoint
- MAP iterations: number of iterations for energy minimization
- Threshold: minimum probability of a pixel to classify it as a lesion

Training datasets are segmented to find the optimal parameters shown in 4.1.

Parameter	Value
K	40
MAP Iteration	10
Threshold	0.90

Table 4.1: Optimal Parameters

4.3 Optimal Parameters

4.3.1 Features

KNN requires a feature extraction process for its memorized training set. The performance of KNN classifier relies on these extracted features. Additionally, the distance metrics play a small role on the accuracy of the labelling process. Therefore, the chosen features must be able to produce the optimal segmentation using KNN.

Intensity

The training dataset we used includes three sequences: FLAIR, T1, and T2 for each patient. In this experiment, we aim to have an idea of the intensity separation of lesion and non-lesions in each sequence. Figure 4.1 to figure 4.6 show the intensity box-and-whisker plots of 6 different MRI scans. The intensities are normalized between the range $[0, 1]$. This gives us a generalized intensity plot among the three sequences. We used the CHB rater as our ground truth in all experiments for it is present among all training sets.

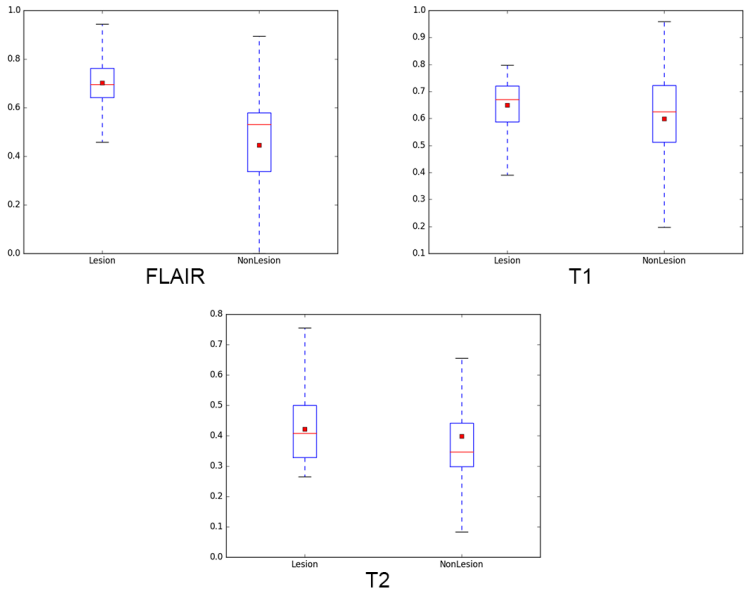

Figure 4.1: CHB Case 01 - Skullstripped, Normalized

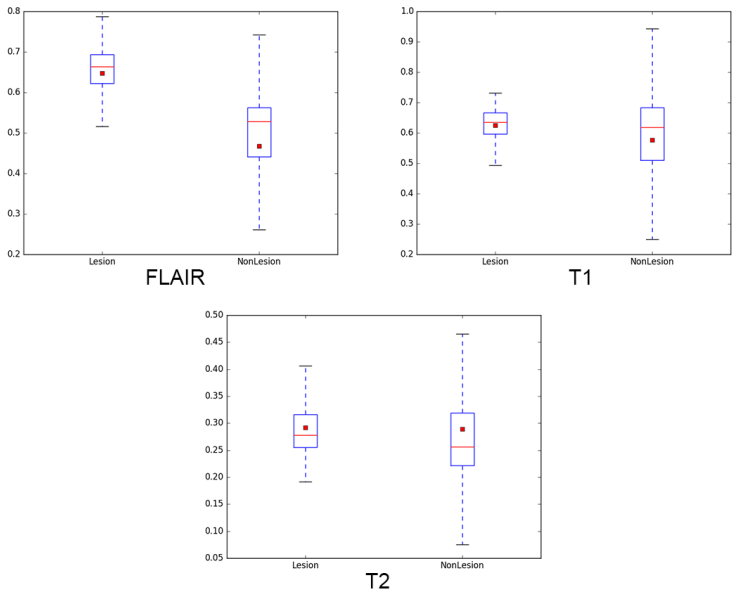

Figure 4.2: CHB Case 02 - Skullstripped, Normalized

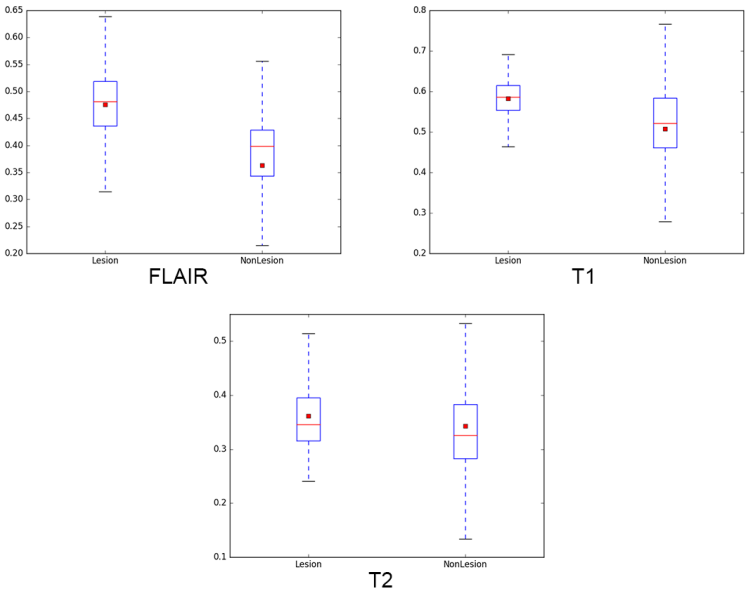

Figure 4.3: CHB Case 03 - Skullstripped, Normalized

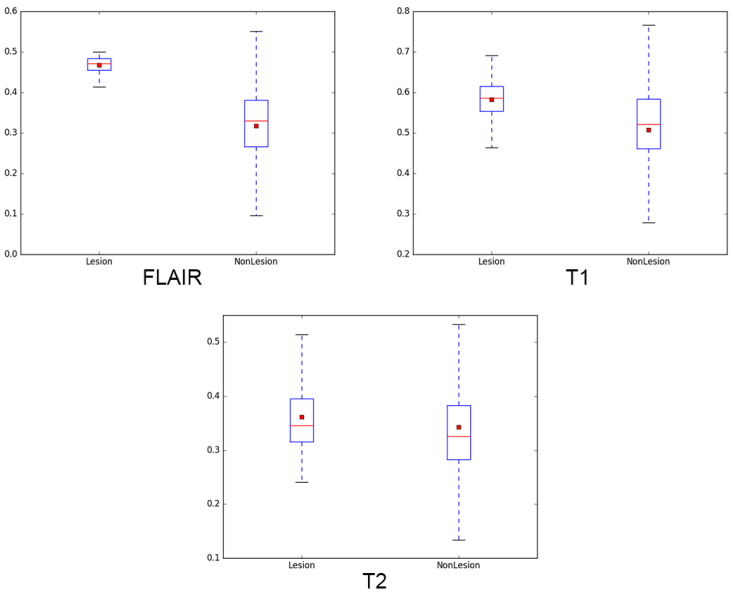

Figure 4.4: UNC Case 01 - Skullstripped, Normalized

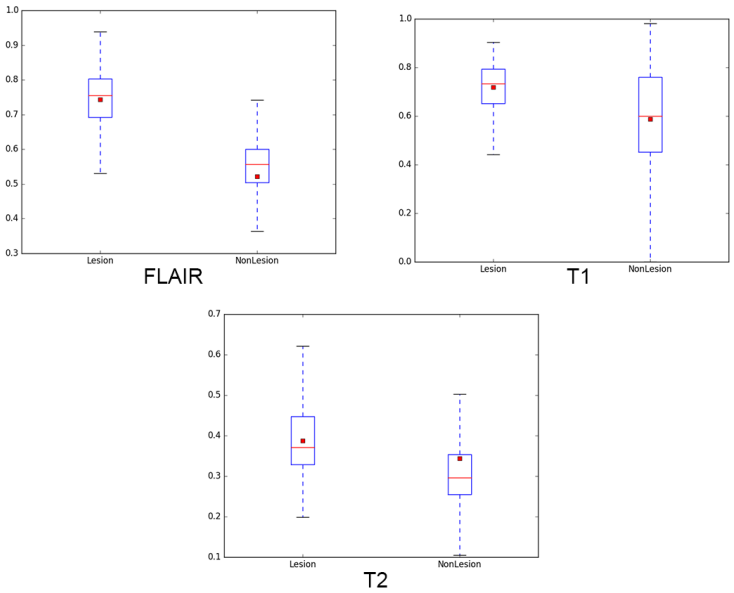

Figure 4.5: UNC Case 02 - Skullstripped, Normalized

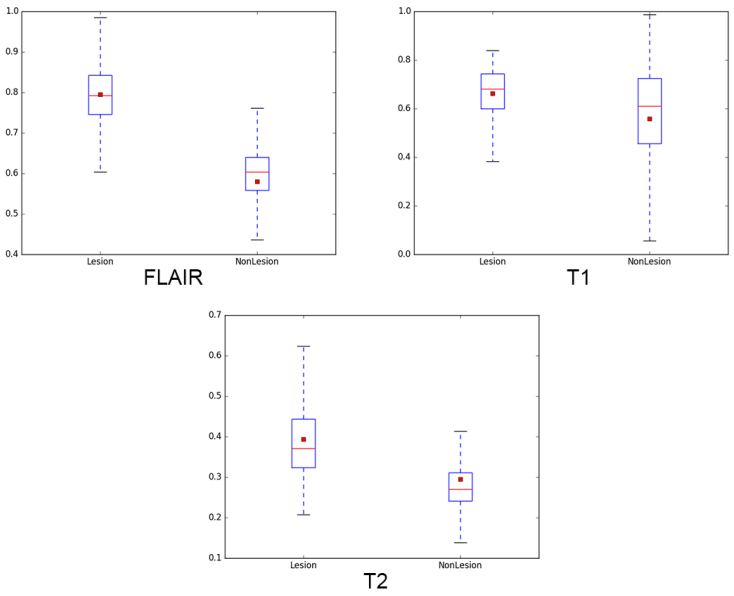

Figure 4.6: UNC Case 03 - Skullstripped, Normalized

As seen in the plots, FLAIR has the most observable intensity separation between lesion and non-lesion while T1 comes in second. T2, however, shows no sign of separation thus it is discarded for this research study. It is seen that T1 is too unstable as a lone feature, however, it has potential. For experimentation, T1 is combined with FLAIR.

Spatial Position

It is visually seen that MS lesions tend to appear in specific regions in the brain. We use scatter plots to statistically observe if there is a pattern in the spatial coordinates of MS lesions. Since spatial information alone doesn't give any reliable insight on the lesion-likelihood of a pixel, it is used alongside with the intensity features.

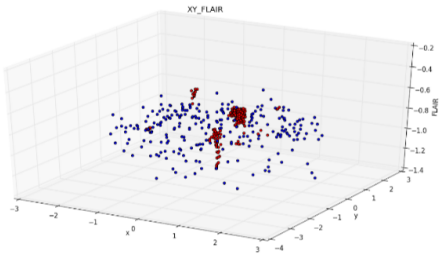

Figure 4.7: Scatter Plot of x, y, FLAIR

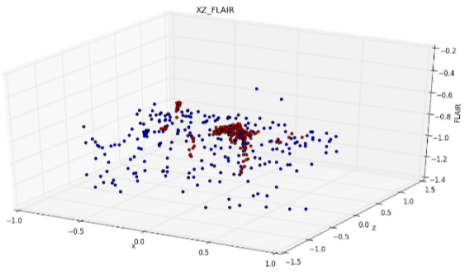

Figure 4.8: Scatter Plot of x, z, FLAIR

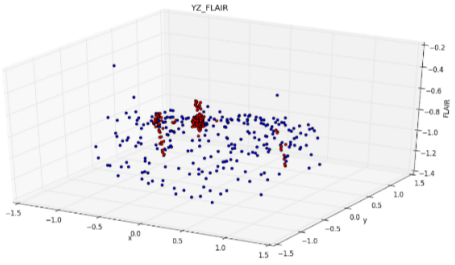

Figure 4.9: Scatter Plot of y, z, FLAIR

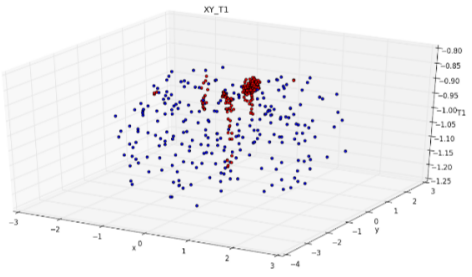

Figure 4.10: Scatter Plot of x, y, T1

It is observed that in spatial-FLAIR information, lesion points (depicted by the red dots) tend to clump together in a specific region. The T1 scatter plot, although showing the same behavior, more lesion points tend to diverge from the lesion regions. From these observations and as mentioned in [9], spatial-FLAIR and spatial-FLAIR-T1 are possible reliable feature combinations for the memorized training set of KNN.

Feature Combination

In this experiment, we aim to find out the optimal features to use for KNN. We made an increasing k experiment for the analysis of the 4 different feature combinations. Figure 4.11 shows an example of the results of the KNN classifier evaluated by four different metrics.

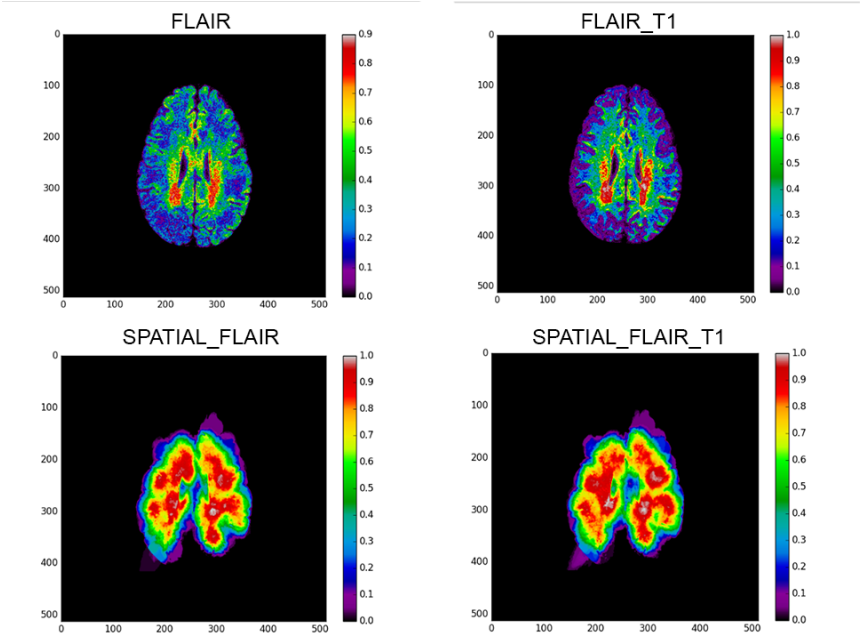

Figure 4.11: KNN Sample Results among 4 Feature Combinations. Red to white are considered to be lesion regions of high probability.

Since results of KNN are probabilistic, we introduced a threshold of 85% for

the validation of our results. This means a probability greater than or equal to the threshold means that the pixel in question is a lesion. The metrics we used are defined as follows:

- Matthew's Correlation Coefficient (MCC): known to be one of the best used metrics to determine the quality of a binary classifier. Unlike other binary classifier metrics, such as the F1 score, this takes into account the true/false positives and negatives. This is generally regarded as a balanced measure which can be used even if the classes are of very different sizes. The return value ranges from -1 to 1, from having a total disagreement between prediction and observation to having a perfect prediction.
- Precision: known as the positive predictive value. This is the total number of automatic segmented lesions that agree with the ground truth over the total population of automatic segmented lesions.
- Accuracy: the total of right predictions over the population.
- Miss Rate: also known as the false negative rate. This is the total number of lesion in the ground truth that isn't present in the automatic segmented lesions over the lesion population of the ground truth.

$$M_c = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{4.3.1}$$

Figure 4.12 shows the trends of the four feature combination of each metric. From the figure, we get spatial information and FLAIR as the optimal feature combination. It not only has the highest ratings in precision and MCC but also has the lowest miss

rate. Thus, spatial-FLAIR is used for the feature combination. The optimal k can be extracted from this experiment. Maximizing MCC, we got an optimal k value of 40.

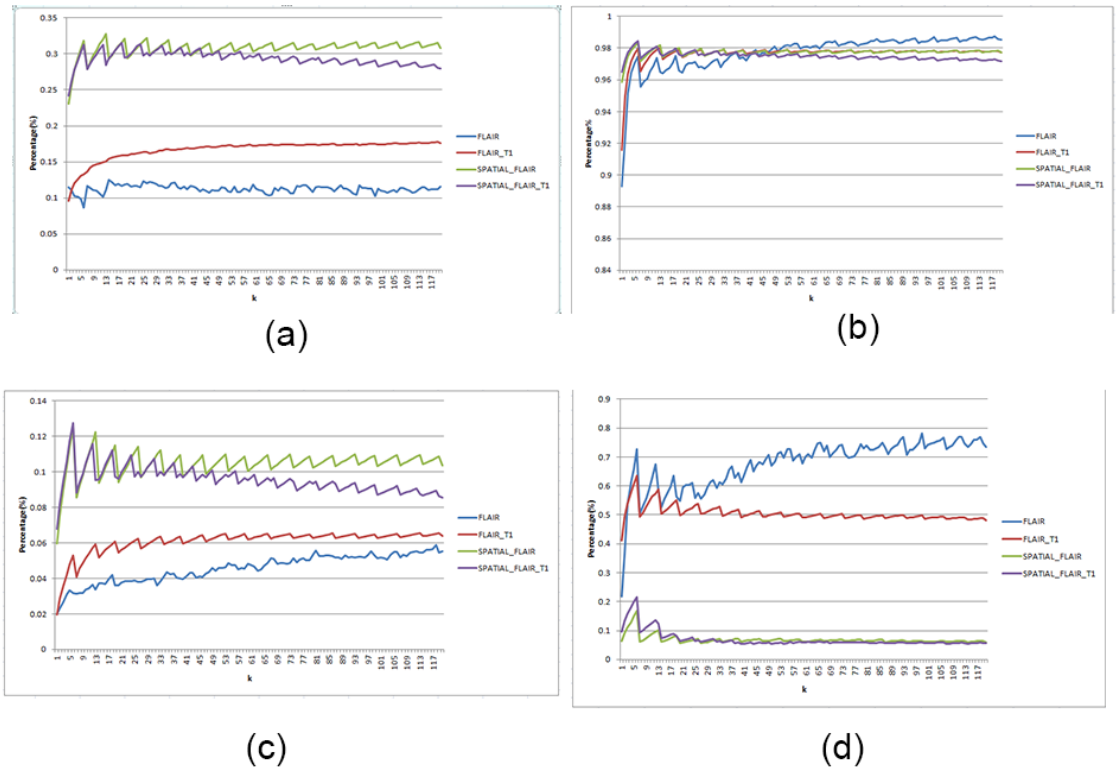

Figure 4.12: The Different Trends of the Evaluation Methods: (a) MCC, (b) Accuracy, (c) Precision, (d) Miss Rate in 4 Features

4.3.2 Maximum A Posteriori (MAP) Iteration

The objective of MRF is energy minimization via local Markov property. Using the local tissue information in the initially segmented brain MRI, we gain more neighboring pixels that show agreement with each other, thus lowering the energy value of the neighborhood. A series of MAP iterations is performed to minimize the global energy (sum of energy values of all neighborhoods).

Three probabilistic MRI scans with different MAP iterations and the ground truth

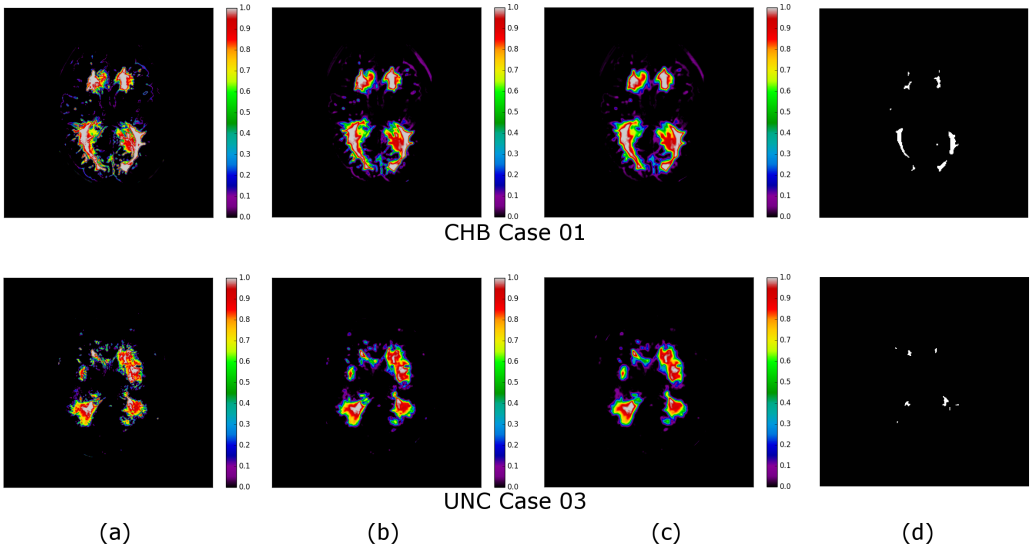

Figure 4.13: Probabilistic Segmentation of 2 Training Scans with (a) 1, (b) 5, (c) 10 MAP Iterations

from 2 training datasets are shown on figure 4.13. Based on visual inspection, both the probabilistic scans with 10 MAP iterations show the most promising probabilistic segmentation. Thus, we chose to use 10 MAP iterations for our study.

4.3.3 Threshold

A threshold is applied to obtain a binary segmentation from a probabilistic segmentation. A threshold value denotes the minimum probability of a pixel to be classified as a lesion. Since BORKOV contains two main processes, namely KNN and MRF, and both produce probabilistic segmentations, two threshold values are needed.

For KNN, we followed its "majority voting" principle, thus the threshold for KNN is automatically the majority percentage, specifically 52.5% at k = 40. For MRF, based on figure 4.13c, the promising threshold is about 90%, which we eventually chose for our experiments.

4.4 Experimental Results

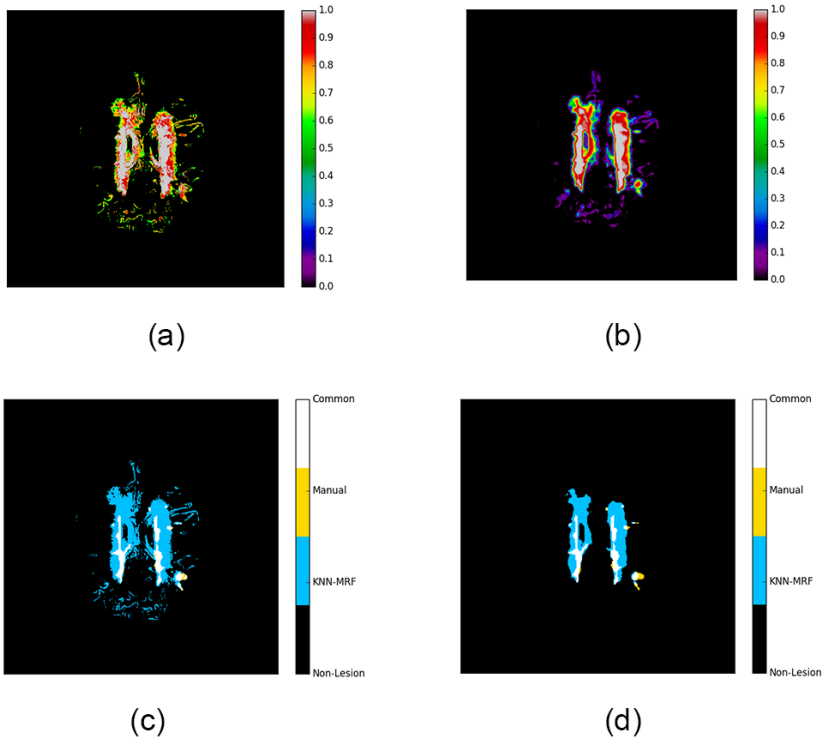

Figure 4.14: CHB Case 08 slice 250: (a) KNN Output, (b) After MRF Output, (c) KNN Compared with Manual Segmentation, (d) KNN-MRF Compared with Manual Segmentation

In figure 4.14, both KNN and KNN-MRF segmentations used the same parameters. KNN output alone introduces a large number of false positives and random thin cotton-like probable lesion regions outside the true lesion regions. Thus, as the figure has shown, MRF optimizes the initial KNN output via local Markov property, reducing the noisy areas.

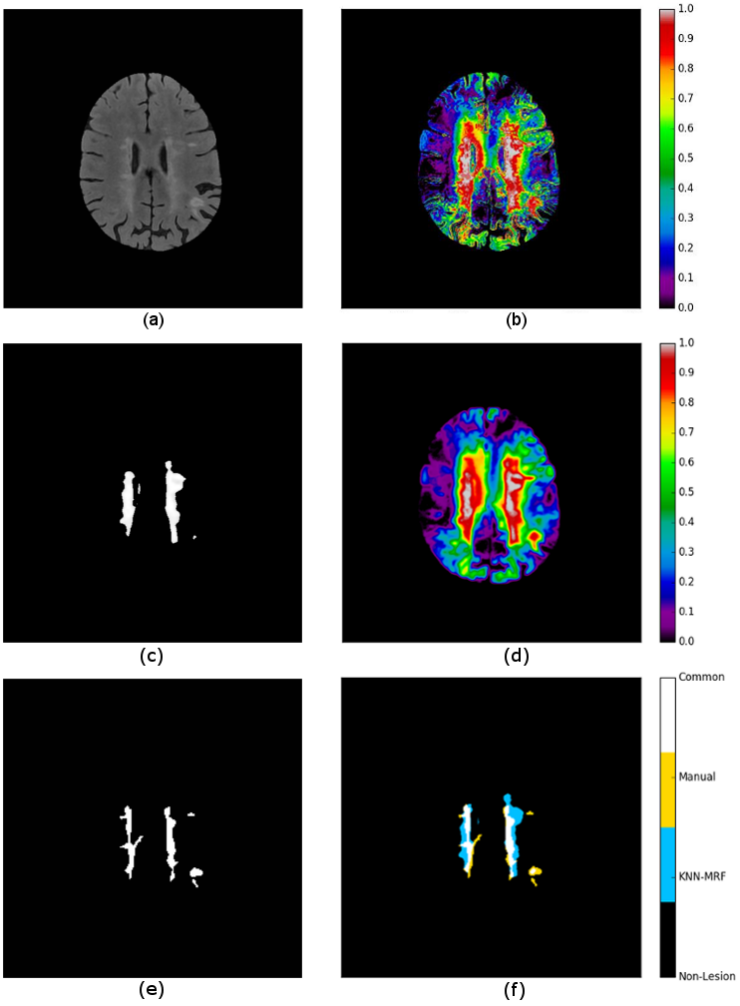

Figure 4.15: CHB Case 08 slice 260: (a) skullstripped, (b) probabilistic KNN segmentation, (c) labelled BORKOV segmentation, (d) probabilistic BORKOV segmentation, (e) manual segmentation, (f) comparison of BORKOV segmentation and manual segmentation

BORKOV is performed using six test datasets, three from CHB and three from UNC. The results and comparison of our automatic segmentation are presented in figure 4.15 with the optimal parameters specified in table 4.1. The probabilistic result of KNN is shown in figure 4.15b. Some areas of this initial probabilistic segmentation show image noise. This is denoised under MRF, as shown in figures 4.15c and 4.15d,

the labelled segmentation and the probabilistic segmentation of BORKOV, respectively. The comparison of our labelled segmentation and the manual segmentation by the CHB rater (figure 4.15e) is shown in figure 4.15f.

	TPR	TNR	PPV	FDR	FNR	Accuracy	MCC
CHB08	0.7249	0.9977	0.2595	0.7404	0.2750	0.9974	0.4328
CHB09	0.6705	0.9980	0.1172	0.8827	0.3294	0.9979	0.2798
CHB10	0.9697	0.9959	0.0591	0.9408	0.0302	0.9959	0.2389
UNC08	0.3042	0.9991	0.0311	0.9688	0.6957	0.9991	0.0971
UNC09	0.6227	0.9990	0.0634	0.9365	0.3772	0.9989	0.1985
UNC10	0.4113	0.9994	0.0961	0.9038	0.5886	0.9993	0.1985
All Average	0.6172	0.9982	0.1044	0.8955	0.3828	0.9981	0.2409
All CHB	0.7884	0.9972	0.1452	0.8547	0.2115	0.9971	0.3171
All UNC	0.4460	0.9992	0.0635	0.9364	0.5539	0.9991	0.1647

Table 4.2: Performance of BORKOV in the 6 Datasets

Table 4.2 shows how our method performs in the six test datasets. The manual segmented scans of the CHB rater are used as ground truth to validate our results. This is because they show better resemblance to the automatically segmented lesions and are available in all testing datasets. The metrics used for evaluation are as follows:

- True Positive Rate (TPR): also called sensitivity; the number of automatically segmented lesions that agree with the manual segmentation over the total number of manually segmented lesions
- True Negative Rate (TNR): also called specificity; the number of classified non-lesions by BORKOV that agree with the manual segmentation over the total number of classified non-lesions by the observer
- Positive Predictive Value (PPV) or Precision
- False Discovery Rate (FDR): the number of automatically segmented lesions

that do not agree with the manual segmentation over the total number of automatically segmented lesions

- False Negative Value (FNR): calculated as $1 - TPR$
- Accuracy
- Matthew's Correlation Coefficient (MCC)

BORKOV performs significantly better using the CHB datasets compared to using the UNC datasets since the UNC MRI scans have the smaller lesion regions among the test datasets. Thus, our method performs best on MR images with big lesion regions.

Accuracy and TNR are high since most pixels are background pixels, thus easily classifying them as non-lesions. Our method produces a high FDR but also a high TPR because our threshold values mean having a wider area of lesion segmentation. Having a higher threshold would mean low FDR and TPR.

The comparison of BORKOV to other segmentation methods is shown in table 4.3. The other significant segmentation methods are as follows: Hierarchical Markov Random Fields and Random Forest Segmentation (HMRF-RFS) [4], Non-Local Means Inpainting (NLMI) [8], Population Intensity Outlier model (PIO) [14], and K Nearest Neighbors only (KNN-only) [9].

We can see that automatic segmentation methods produce either a high TPR and FDR (oversegmentation) or a low TPR and FDR (undersegmentation). Comparing our results with the KNN-only method, it still can be seen that a high TPR rate is compromised with a high FDR rate. BORKOV produces the highest FDR rate, but also the highest TPR rate among the five methods.

It is important to note that the other methods in table 4.3 used more testing

	Average		All UNC		All CHB	
Methods	TPR	FDR	TPR	FDR	TPR	FDR
HMRF	53.5%	24.2%	74.4%	43.7%	40.0%	11.6%
NLMI	52.7%	42.0%	63.1%	53.2%	46.0%	34.7%
BORKOV	61.7%	89.5%	44.6%	93.6%	61.6%	82.6%
PIO	51.8%	45.1%	62.3%	54.7%	45.0%	38.9%
KNN-only	59.1%	78.5%	51.4%	67.3%	64.7%	86.5%

Table 4.3: Comparison of BORKOV to Other Segmentation Methods

datasets that are different from our testing datasets. If their testing datasets contain more large MS regions, BORKOV will perform significantly better.

It was a limitation that the only data available to be used as features are FLAIR, T1, T2, and spatial information. Having more features as data could mean a better segmentation if and only if that feature has good quantifiable separation of lesion and non-lesion tissue. According to [15], about radiology protocols on MS lesions, they included ost-single-dose gadolinium-enhanced T1-weighted sequences and a DWI sequence. Spatial information introduced false positives in the areas where lesions mostly occur. This could mean FLAIR intensity is not enough of a feature to influence the lowering of the lesion probability among those regions. The quality of the training datasets is also a big factor for KNN. We were also limited to 14 training datasets.

It is a fact that lesion tissues have their intensities gradually incline to intensities of normal tissues, therefore posing a problem to automatic segmentation. As far as our study is concerned, there are no definite distinct features yet that separate lesion to non-lesion tissues.

It was discussed in [11] that there's a problem with choosing the right ground truth: the raters. We believe that there is no ground truth, and manual segmentation used for evaluation is subjective to the radiologist who segments the lesions. This means

that a disagreement between the predicted segmentation and manual segmentation may not always mean a bad performance of the automatic segmentation.

Below is the breakdown of the time duration of each process in BORKOV. The main factor of the time duration is the input size, having a 512x512x512 dimensionality.

Process	Time
Brain Mask Creation	3 mins
Initialization	10 secs
KNN-Proper	5 mins
MRF-Proper	32-55 mins
Results Creation	4 mins
Total	44-67 mins

Table 4.4: Time Duration of BORKOV

Chapter 5

Summary and Conclusion

In this paper, we present BORKOV, an automatic MS lesion segmentation using a two-step approach: KNN and MRF. Our fully reproducible method produces good enough results to be suitable for usage in clinical practice and in longitudinal cohort studies. The automatically segmented lesions are comparable to manually segmented lesions.

BORKOV performs best using MR images with high lesion regions compared to using images with small lesion regions. It produces high accuracy and acceptable lesion detection rate. Just like in other automatic segmentation methods, undersegmentation and oversegmentation pose a problem for our method. These are due to the limited feature information in the memorized training set of KNN and to the local Markov property of MRF. However, in large-scale cohort studies, undersegmentation and oversegmentation may not be considered an issue as long as these are a consistent happening.

The probabilistic result of BORKOV poses an innovation to medical image processing since most segmentation methods have labelled segmentation as their output. The problem with labelled segmentation is that it forces a certain pixel to have a

particular label, even if its probability is barely enough to classify it as the specific label. A probabilistic segmentation gives way to flexibility; it acquires a user specified confidence level percentage or threshold for the lesion segmentation. This addresses the issue of undersegmentation or oversegmentation (but not both).

Appendix A

Description of the Environment and Source Codes

A.1 Environment

The programs runs in the Python environment only. We used specific modules that are available in Python – Scikit-Learn, Matplotlib, and SimpleITK are modules that are indispensable in the implementation of BORKOV.

A.2 Interface Modules

A.2.1 Source Codes

The following modules of the program serve as the interface of the software.

Algorithms.py

This contains the commonly used functions among our source codes. This contains six functions.

loadMRI This is used to load the MR image into a SimpleITK image object. This needs a filename input.

convertToArray This is used to convert the SimpleITK image object into a NumPy array of values. This needs a SimpleITK image input.

normalize This is used to normalize data into the range [0,1]. This needs the raw data, min, and max as inputs.

standardize This is used to standardize data or convert data into their corresponding z-scores. This needs the raw data, mean, and std as inputs.

readTrainingData This is used to parse the training data input from KNN. This requires a filename as input. This will input a multi-dimensional jagged array containing features, labels, technique, min/mean, and max/std.

createBrainMask This is the implementation of the brain mask algorithm mentioned above. This returns a 2d array indicating the slice and box coordinates of the created brain mask.

KNN_Train.py

This is the training module of the creation of the KNN memorized training set. It has the methods of the five processes as functions. It also has parameters such as -o for options, 1 - FLAIR only, 2 - FLAIR and T1, 3 - Spatial and FLAIR, and 4 - Spatial, FLAIR, and T1, -t for how many training datasets will be inputted, and -p for the percentage of the population you want randomly selected.

memorization This needs a T1 MRI input. This creates the memorized coordinates of positive brain tissue as a text file.

segregation This needs the ground truth and the corresponding memorized text file. This will split the memorized text file into two separating non-lesions and lesions. This is also equipped with a random sampler for the bias correction problem.

retention This needs the segregated text files. This is simply a random sampler further reducing the population of the lesion and non-lesion.

chunking This is the feature extraction function of the training module. This needs the retention text files as input combining into one .csv file.

standardization Using NumPy, we can get the mean and standard deviation of each feature in the dataset. This creates the final standardized memorized training set.

Main.py

This is the testing module incorporating both the KNN testing module and the MRF module. This is the core program for the automatic segmentation of the MR image. This has parameters such as : -k the number of neighbors in KNN, -t the threshold as described in the research, -i the number of MAP iterations.

knn This needs the k, FLAIR, brain mask and memorized training set as inputs. This returns a probabilistic output. The classifier created is set to do n parallel jobs, n is the number of the CPU cores. This is automatically computed using the multiprocessing module of Python.

mrf The MRF module. The process is a per slice basis.

MAP Energy function is minimized by finding the optimal probability of a pixel.

energyFunction It computes the sum of all values of pairwise affinities in a given neighborhood.

affinityFunction It measures the similarity of the discrete probability distribution of two neighboring pixels.

getNeighbors Returns the neighboring pixels of the pixel of interest.

knn-mrf This is the method that combines both modules. This includes the thresholding and saving of the output. This needs the k, threshold, and MAP iteration as inputs from the parameters passed in the command prompt.

Comparison.py

Utilizing the Matplotlib module of Python, this produces the comparison of the performance of BORKOV to the corresponding ground truth of the inputted dataset.

A.2.2 Python Modules Used

Scikit-Learn

The KNeighborsClassifier class of the Scikit-Learn module was used. This module enabled us to create a KNN classifier with user specifications of k. It also has a function that can return the probabilities of the pixel under the specified classes. In this case, the probabilities returned were probability of being a non-lesion and a lesion. The lesion probability is on the second index.

Matplotlib

This is used widely in creating representations of data (charts/graphs) and can be used for image manipulation.

SimpleITK

This is used to load the MR images for manipulation.

multiprocessing

This is used to scan for the machine's number of cores.

random

This is the random sampler module used in the training process.

Tkinter

This is used to create a file chooser dialog box asking for the required inputs.

scipy.misc

This is used to save the probability array of each slice of an MRI to a .png file.

Bibliography

[1] Multiple sclerosis diagnostic tests. http://www.webmd.com/multiple-sclerosis/guide/multiple-sclerosis-diagnosis-tests.

[2] National multiple sclerosis society. http://www.nationalmssociety.org.

[3] Omar Islam. Brain magnetic resonance imaging technique, Aug 2015. http://emedicine.medscape.com/article/2105033-technique.

[4] Andrew Jesson and Tal Arbel. Hierarchical mrf and random forest segmentation of ms lesions and healthy tissues in brain mri. 2015.

[5] Dominique Zosso Jean-Philippe Thiran Marie Schaer, Laurent Guibaud. Brain surface segmentation of magnetic resonance images of the fetus. *16th European Signal Processing Conference*, Jan 2008.

[6] Pierre-Marc Jodoin Mohammad Havaei and Hugo Larochelle. Efcient interactive brain tumor segmentation as within-brain knn classification. *Universite de Sherbrooke*, pages 1–6, Dec 2013.

[7] S. J. Francis S. Narayanan D. L. Collins D. L. Arnold N. K. Subbanna, M. Shah and T. Arbel. Ms lesion segmentation using markov random fields. *Work. Med. Image Anal. Mult. Scler.*, pages 15–26, Jan 2009.

[8] Pierrick Coupé Vladimir S. Fonov Douglas L. Arnold Nicolas Guizard, Kunio Nakamura and D. Louis Collins. Non-local means inpainting of ms lesions in longitudinal image processing. *Frontiers in Neuroscience*, 9(456), 2015.

[9] Koen L. Vincken Petronella Anbeek and Max A. Viergever. Automated ms-lesion segmentation by knearest neighbor classification. *Insight Journal*, pages 1–8, Jul 2008.

[10] Ann Pietrangelo and Valencia Higuera. Multiple sclerosis by the numbers: Facts, statistics, and you, Mar 2015. http://www.healthline.com/health/multiple-sclerosis/facts-statistics-infographic.

[11] Raul Rojas. Nearest neighbors classifiers. *Freie Universitat Berlin*, pages 1–7, Jul 2014.

[12] Stephen Smith. Fast robust automated brain extraction. *Human Brain Mapping*, 17(3):143–155, Nov 2002.

[13] The Heathline Editorial Team. Early signs of multiple sclerosis, Nov 2015. http://www.healthline.com/health/multiple-sclerosis/early-signs1.

[14] X. Tomas-Fernandez and S. K. Warfield. Population intensity outliers or a new model for brain wm abnormalities. *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1543–1546, May 2012.

[15] James A Wilson. Brain imaging in multiple sclerosis, Mar 2016. http://emedicine.medscape.com/article/342254-overview.

[16] J. Ross Quinlan Joydeep Ghosh Qiang Yang Hiroshi Motoda Geoffrey J. McLachlan Angus Ng Bing Liu Philip S. Yu Zhi-Hua Zhou Michael Steinbach David J. Hand Dan Steinberg Xindong Wu, Vipin Kumar. Top 10 algorithms in data mining. *Springer-Verlag London Limited 2007*, pages 22–24, Dec 2007.