A Deep-Learning Convolutional Neural Network Framework for Multiple Sclerosis Lesion Detection and Segmentation in Patient Brain Images

Maor Zaltzhendler

Department of Electrical & Computer Engineering
McGill University
Montréal, Canada

November 2015

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering.

THIS PAGE INTENTIONALLY BLANK

Abstract

This thesis presents a convolutional neural network (CNN) based approach for detection and segmentation of Multiple Sclerosis lesions in brain magnetic resonance imaging (MRI). Automated pathology segmentation was presented in literature, starting from the early 1990s, and although reported to be a challenging task, could be highly beneficial for clinical trials labeling where large amounts of images are at hand. Robust detection of such pathology is still an open problem, and is prone to variabilities in: image non-uniformity, intensity distributions, acquisition artifacts, brain-structures, patients, scanners, configurations and sites. A CNN-based approach is proposed due to its recently reported high quality and generalization properties for computer vision tasks, providing a high degree of invariance and taking spatial correlation within the image structure into account. In order to address the task using both local and context-related information, a multi-scale approach is suggested, integrating the accuracy of several CNNs within a hierarchical framework for pathology segmentation. The presented model is general, and could be used for other pathology detection and segmentation contexts that require object delineation and classification in 3D magnetic resonance imaging. Several different architectures and experiments are presented throughout the document, while providing benchmarks and qualitative views over their results. Additional contributions of this thesis include: (a) learning CNN-based brain-features, evaluating their discriminative power, and observe appearance and constancy, (b) develop a general approach for MRI segmentation, while naturally incorporating the full 3D neighbourhood information rather than using 2D or augmented-2D with consecutive slices information. A comprehensive set of experiments is provided throughout this thesis, and performed over two different multi-site large scale proprietary clinical trials that were made available for this research. First, the method was configured and tested over the first clinical trial only. Once the hyper-parameters were set, no further tuning was allowed and the architecture was tested over the second clinical trial, which is much larger, and showed similar performance. The results of the method over this data yielded sensitivity values of up to 0.68, and Dice

scores up to 0.59. The method achieved even higher metric scores of 0.86-1.00 true-positive rates when considering only larger lesions. The experiments performed show comparable performance to previously reported results from the literature over the same dataset. The data-driven features are presented, and shown to capture brain structures that lead to MS lesion discrimination both qualitatively and quantitatively.

Résumé

Cette thèse présente une approche basée sur les réseaux de neurones à convolution (RNC) servant à la détection et à la segmentation de lésions cérébrales liées à la sclérose en plaques (SEP) en utilisant l'imagerie par résonance magnétique (IRM). La segmentation automatique de la pathologie est présente dans la littérature depuis le début des années 1990. Cette procédure, bien qu'elle soit difficile à appliquer, pourrait être bénéfique dans les essais cliniques lorsqu'une grande quantité d'images sont disponibles. La détection de cette pathologie demeure un problème et est sujette à des variabilités : non-uniformité des images, distribution d'intensité, artéfacts d'acquisition, structures cérébrales, patients, appareils, configurations et sites. L'approche proposée dans cette thèse est basée sur les réseaux de neurones à convolution (RNC) qui ont récemment démontré des propriétés de haute qualité et de généralisation pour des tâches de vision par ordinateur en offrant un degré élevé d'invariance et en prenant des corrélations spatiales au sein de la structure de l'image. Afin d'exécuter la tâche en utilisant des informations à la fois locales et liées au contexte, une approche multi-échelle est suggérée, en intégrant l'exactitude de plusieurs réseaux de neurones à convolution (RNC) dans un cadre hiérarchique pour la segmentation de pathologie. Le modèle présenté est général et pourrait être utilisé dans d'autres contextes de détection et de segmentation qui nécessitent la délimitation des objets ou de la pathologie dans l'imagerie par résonance magnétique 3D. Plusieurs architectures et expériences différentes sont présentées dans ce document, tout en fournissant des repères et des points de vue qualitatifs sur leurs résultats. De plus, cette thèse comprend des contributions supplémentaires telles que (a) l'extraction de caractéristiques basée sur les réseaux de neurones à convolution (RNC), l'évaluation de leur pouvoir discriminant et l'observation de leur apparence et leur constance, (b) le développement d'une approche générale pour la segmentation de l'IRM, tout en intégrant naturellement des informations de voisinage 3D plutôt qu'en utilisant des images 2D qui incluent des informations sur les tranches consécutives. Cette thèse inclut des expérimentations sur deux essais cliniques privés multi-sites à grande échelle qui ont été mis à la disposition pour cette recherche.

D'abord, la méthode a été configurée et testée sur le premier essai clinique seulement. Une fois que les hyper-paramètres ont été définis, aucun ajustement supplémentaire n'a été effectué. L'architecture a été testée au cours du deuxième essai clinique, qui est beaucoup plus grand, et qui a montré des performances similaires. Les résultats obtenus ont donné des valeurs de sensibilité allant jusqu'à 0,68 et des scores Dice jusqu'à 0,59. Lorsque l'on considère seulement les lésions plus importantes, la méthode atteint des taux vrais positifs entre 0,86 et 1,00. Les expériences réalisées démontrent des performances comparables aux résultats récemment rapportés dans la littérature sur un même ensemble de données. Les caractéristiques obtenues par les données sont présentées de façon qualitative et quantitative. Elles permettent de capturer les structures du cerveau qui conduisent à la discrimination des lésions causées par la sclérose en plaque (SEP).

Acknowledgements

The work presented in this thesis would have not been possible without the support and help of the following friends and colleagues. First, I would like to thank my advisor, Prof. Tal Arbel for our long fruitful discussions that inspired the design and the development of the method. Her guidance throughout the whole process was important for me, and lead this research to cover particularly interesting topics in the field of medical image analysis. Second, I would like to thank Dr. Meltem Demirkus for her incredible creativity and contributions to the presented method, as well as to the thought process that brought it to life. It would have definitely not been possible without her help. I would like to also thank Andrew Jesson from the Probabilistic Vision Group in McGill for his help and very interesting long discussions. Thanks to Prof. Kaleem Siddiqi and Krys Dudek for the interesting experience in the Natural Science and Engineering Research Council Collaborative Research and Training Experience in Medical Image Analysis (NSERC CREATE-MIA) program. I am very thankful for the support from Prof. Aaron Courville and Dr. Nicolas Ballas from the LISA lab in University of Montreal. I would like to express my gratitude to Jan Binder and Nick Wilson for their excellent computing advices and continuous support since my first day in the PVG lab. Also, I would like to thank Prof. Doina Precup for her excellent explanations and sharing of her machine learning expertise, and the time given for discussing potential techniques and ideas. Thanks to NeuroRx Research and Dr. Douglas L. Arnold for providing a very large amount of clinical data, and also generously providing me with a state-of-the-art machine that was used for this research. Thanks to Christopher Murtagh for his incredible IT support during the process. I would like to thank all the members of the Probabilistic Vision Group (PVG) in McGill for being there, brainstorming, presenting interesting topics in our group meetings, and providing me with very helpful comments and thoughts. Last but not least, I would like to thank my wonderful wife, family and friends for their support on the personal level. I was definitely glad having you during this intensive research period.

Contents

List of Figures

List of Tables

List of Acronyms

MS	Multiple Sclerosis
CNS	Central Nervous System
GMM	Gaussian Mixture Model
ANN	Artificial Neural Network
MLP	Multi-Layer Perceptron
CNN	Convolutional Neural Network
ReLU	Rectified Linear Unit
SIFT	Scale-Invariant Feature Transform
SVM	Support Vector Machine
ROC	Receiver Operating Characteristic
AUC	Area Under Curve
MSE	Mean Squared Error
MRI	Magnetic Resonance Imaging
T1w	T1-weighted
T2w	T2-weighted
PDw	Proton-Density weighted
FLAIR	Fluid-attenuated inversion recovery
ICBM	International Consortium for Brain Mapping
TP	True Positive

FP False Positive

FN False Negative

TN True Negative

BLOB Binary Large Object

Chapter 1

Introduction

The research presented in this thesis addresses the challenge of Multiple Sclerosis lesion segmentation in brain MR images. Current clinical protocol involves manual labelling, a process that is expensive, time-consuming and subject to intra- and inter-rater variability. In order to improve this process, several automatic techniques have been introduced in the literature [10, 32, 44, 45, 49, 60, 72, 78–80, 90, 92, 93], many using statistical models that learn to characterize healthy tissues and/or pathology. Spatial information has been integrated into some models [32, 80] using graphical models or hand-crafted features that further model the relationship between adjacent tissues, and therefore consider local correlations between classes. In previous work, Gabor and other handcrafted features were used to capture texture, context and appearance, while reporting accuracy improvement based on the given metrics for the task [32, 80]. The weakness of such techniques is that the set of input features is pre-set subjectively at design time, and therefore cannot adapt to the actual data. As a consequence, the art of manually handcrafting these features is left to the authors, and introduces bias towards their personal preferences, intuition or competency. The method presented in this thesis overcomes this issue by using machine learning techniques named *deep learning*, that are designed to learn the features from the data instead of manually selecting them. Convolutional neural networks, which are a deep learning technique, have recently gained popularity due to their unbeatable performance on large-scale object recognition tasks [68]. In this thesis we employ CNNs for the lesion segmentation task, and adapt them to the context of MR image segmentation. In the next section, we start by presenting Multiple Sclerosis in general, and follow with an overview of the proposed framework.

1.1 Multiple Sclerosis

Multiple Sclerosis (MS) is a chronic neurological disorder, attacking the central nervous system (CNS). MS can affect vision, hearing, memory, balance, mobility, and coordination, often in a pattern of relapses and remissions [3, 5, 6]. The effects of the disease can be physical, emotional and financial [5]. In 2011, the Canadian Community Health Survey (CCHS) reported that 93,535 Canadians are living with MS [2], and the MS Society estimates approximately 1,000 new cases of MS to be diagnosed every year in Canada [5]. In colder climates, the incidence of MS is higher [4]. There are 2.5 million estimated MS cases worldwide, while rates are higher farther from the equator. The rate of MS in southern US is estimated 57-78 cases per 100,000 people, while the rate in northern states reaches 110-140 per 100,000. The highest risk of having MS is among the population of Northern Europe, and in lowest risk are Native Americans, Africans and Asians. See Figure 1.1 for the global median prevalence of Multiple Sclerosis. [4]

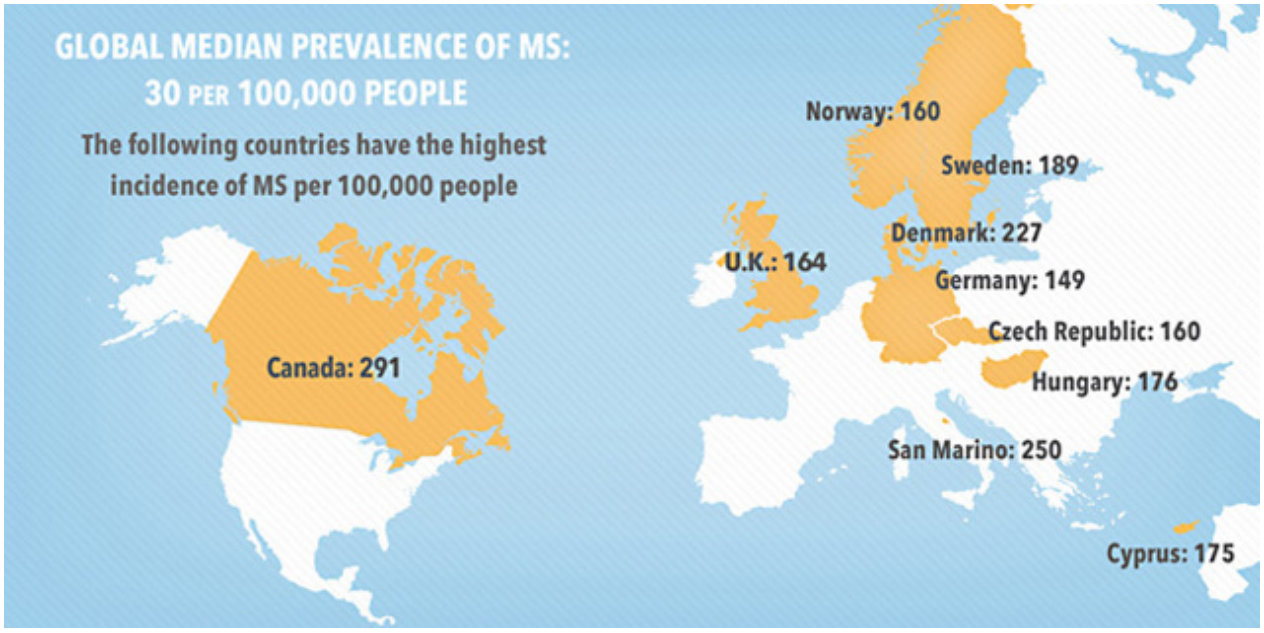


Figure 1.1 Median Prevalence of Multiple Sclerosis, courtesy of [4].

In the context of MS, the immune system attacks the myelin layer that surrounds the axons or nerve fibers and provides insulation. This phenomena is known as demyelination, and appears in multiple sites of the brain and spinal cord comprising the central nervous system. Unfortunately, this process could also affect the ability of the brain to repair by damaging the cells that produce myelin [3]. Figure 1.2 [28] illustrates the demyelination of

axons. The multi-site scarring caused by demyelination disrupts the communication signals used for transferring messages between the brain and other body-parts through the spinal cord. Furthermore, direct damage to the axons resulting from the inflammation could cause a permanent loss of function. An intuitive analogy could be given by depicting an electrical communication wire. Multiple sclerosis attacks the insulation, resulting in fraying and short-circuiting that prevents the correct signal from reaching its destination. [3]

Important criteria for the quantification of multiple sclerosis in clinical trials rely on expert-labeling of the lesions over several different magnetic resonance imaging modalities. The labeling is used for disease diagnosis, progress tracking, or drug influence analysis in clinical trials. Current clinical protocol involves manual labelling, a process that is expensive, time-consuming and subject to intra- and inter-rater variability. Since the definition of Multiple Sclerosis lesion in MRI is imprecise, and generally refers to 3 different categories of lesions (T2-lesion, enhancing-lesion and black-hole), the task of automatic classification and segmentation of such pathology involves experience, or data-driven techniques. Supervised semi- and automatic techniques learn the definition of MS lesion from given expert-labeled examples rather than by using a concise definition of the pathology. The literature covers a broad selection of different techniques, from the early 90s until today, which are driven by machine learning, feature extraction and statistics [29]. Unsupervised techniques, which are the other class of MS lesion segmentation methods, are mainly based on clustering and outlier detectors [10, 30, 74, 78, 87]. Several different MS lesions are shown in Figure 1.3, along with their segmentations. The segmentation task becomes challenging due to the extreme variability in the properties of lesions, such as their size, shape and textures, as well as the overlap of their intensity distributions with those of healthy tissues. Furthermore, lesions can be extremely small, spanning at most a few voxels.

Many of the previous techniques consider the voxel-intensity of the MR modalities as their major features for classification, and build a statistical model that learns the characteristics of healthy tissues versus pathology [36, 44, 45, 49, 60, 90, 92–94]. Other techniques focus on learning only the distributions of healthy tissues, and consider outliers as pathology [10, 29, 30, 74, 78, 87]. Spatial information was also integrated into these models using graphical models, such as Markov Random Fields, or hand-crafted features that further model the relationship between adjacent tissues, and therefore consider local correlations between classes [32, 80]. Context-aware models that learn more than the local properties at the voxel-level reported higher results over public challenges [10, 49, 60, 72, 78, 79, 93].

Many techniques use higher level features for segmentation [32, 80]. For example, previous

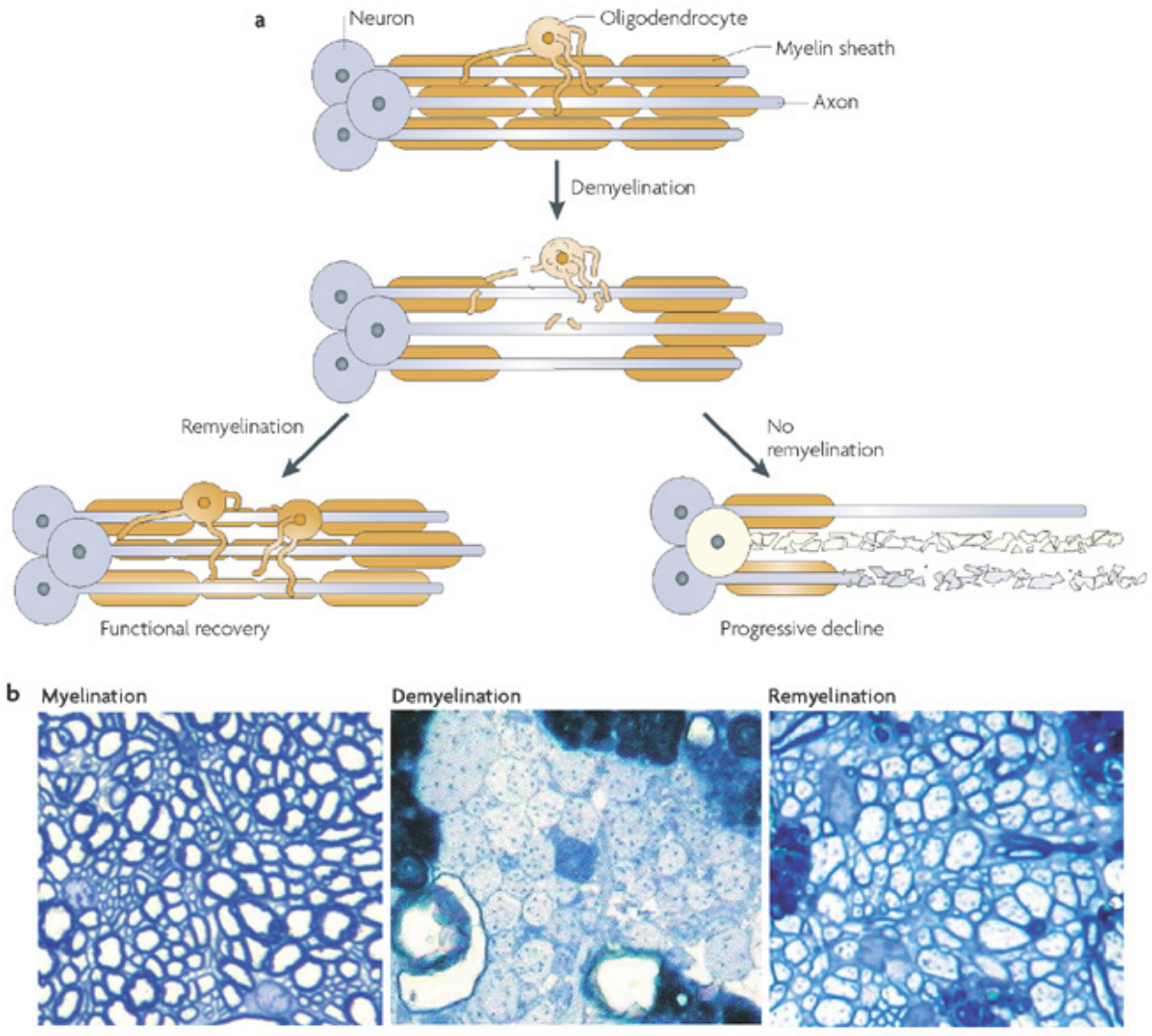


Figure 1.2 Demyelination of axons, courtesy of [28]. (a) An illustration of demyelination in the central nervous system. Once demyelination appears, the sheaths could be recovered in a process called remyelination. This recovery, however, results in thinner sheaths as shown in the left-hand side of the figure. When remyelination fails, such as in multiple sclerosis, the axons remain vulnerable, resulting in degeneration as shown on the right-hand side of the figure. (b) Transverse sections of cerebellar white matter. From left to right: (i) normal axons, (ii) demyelinated axons, and (iii) remyelinated axons.

work [80] used Gabor filters over the MR images in order to capture discriminative features, such as texture and context and use them as features for a more informed model, and their reported results show that these kind of features improve the classification accuracy. Other

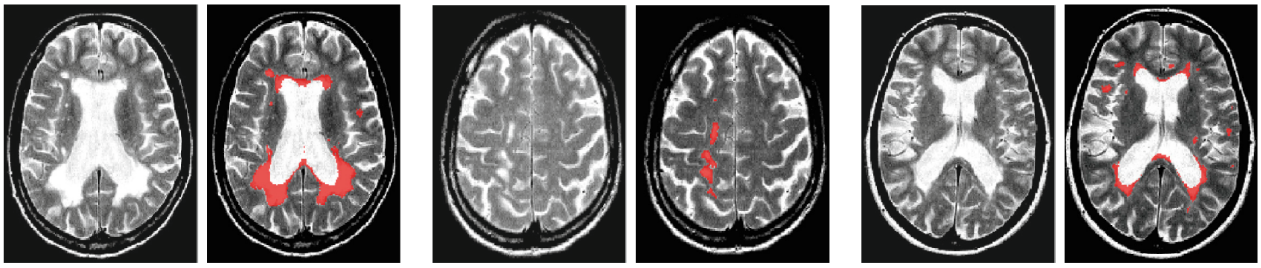


Figure 1.3 Multiple Sclerosis Lesion Segmentation for Three Different Patients, courtesy of [81]. In this figure, we see several kinds of MS lesions and their segmentations (in red). The segmentation task becomes challenging due to the different characteristics of these lesions such as texture, size, intensity and shape. From left to right: (a) high peri-ventricular lesion load, (b) supra-ventricaular lesions, and (c) juxta-cortical lesions.

work [32] used a large amount of different handcrafted context features and a random forest to select them, and also reported improvement in accuracy based on the given metrics for the task. The main weakness of such techniques is that the set of input features is decided at the design time, and therefore cannot be adjusted according to the actual data. As a consequence, the challenge of manually handcrafting these features is left to the authors of the methods, and introduces bias towards their personal preferences, intuition, or competencies. For example, these methods could obtain different results for two different choices of features.

Deep learning approaches in general, and particularly convolutional neural networks (CNNs) learn the features of the input data, whereas classical approaches require manual selection and hand-crafting of features. Manual feature selection is expensive, time consuming and introduces bias. In the field of computer vision, approaches based on deep learning have been obtaining unrivaled results for the task of image classification [38, 39, 48] over recent years, but has not been widely explored for the context of pathology detection and segmentation in medical images (e.g. MS lesion segmentation). An interesting result of applying deep learning for the lesion segmentation task is its custom features set. Since the features are extracted in a fully automatic manner, and without manually designing them, they are expected to become highly adapted to the task and data. The part of the model that generates these features, once trained, could be further used for visualization or better understanding of the structures that lead to the final decision. Another advantage of the proposed approach is that the notion of neighbourhood information rather than local voxel intensity will also be learned from the data. The spatial characteristics of lesions will be

learned from their labeling given a convolutional neural network, that has the capacity to model extremely complex context-aware relationships between different voxels, over different modalities, and learn a data-driven notion of context. This notion of context have been explored in the past by Geremia et al. [32] using a set of context-rich features that compare the voxel of interest to distant regions, measuring symmetry and other neighbourhoods. Another way that was widely used for context-aware methods is through Markov Random Fields (MRFs) [36, 44, 80, 93, 93] to assure local smoothness and neighbourhood awareness by adding spatial connectivity between the voxel classes. In this thesis, I will connect between these two domains, which are machine-learning and medical image analysis, and present a method that uses deep learning for MS lesion segmentation.

1.2 Outline of Framework

In this thesis I will address the task of labeling lesions in multi-modal MR images, such as fluid-attenuated inversion recovery (FLAIR), proton density (PD), T1- and T2-weighted. The output image is typically binary, in which there are two classes: (a) lesion, and (b) non-lesion. The objective is the detection of the lesions in the input images and delineating their boundaries. The objectives of this thesis is to develop a framework for the context of analysis of clinical trials, that is one that is robust to variability stemming from patient brain images acquired from different scanners, centers, and trials worldwide, and for patients at different disease stages, while retaining accuracy, particularly at detection of very small lesions. The method is required to be accurate for large datasets acquiring during different clinical trials. A high-level view of the proposed method is shown in Figure 1.4.

For the first component, a multi-scale approach is proposed in order to model the input image and obtain features that describe different characteristics of each voxel. This component is based on 3 different convolutional neural networks in order to represent the input data. The framework builds a local voxel-intensity model using the first CNN, while also providing it with spatially local neighbourhood information to promote robustness to noise. The voxel intensity model is designed to not carry any notion of context or neighbourhood information, and is intended to learn only how lesions would appear through a small aperture. The second component, also based on a CNN, is a higher-scale component of the close-neighbourhood information, which was designed to describe how nearby voxels appear and gain more contextual information rather than intensity. The last scale component in the model is designed to characterise a very-large neighbourhood, and mostly provide context information rather

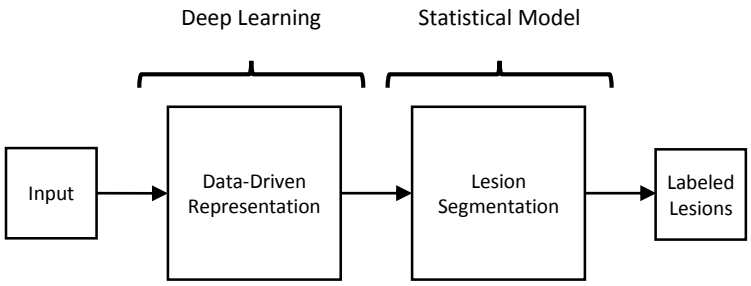


Figure 1.4 High-Level Method Flowchart. The input data is transformed into a deep data-driven feature representation by using 3 Convolutional Neural Networks (CNNs) for creating a multiscale representation of the MR image. Several approaches are used for the statistical model for the joint distribution, including Artificial Neural Networks (ANNs), Random Forests (RFs) and Convolutional Neural Networks (CNNs). The output of these models is the final segmentation result.

than local information. This multi-scale approach creates a separation between the different scales and reduce co-adaptation of their data-driven features. As will be discussed in details throughout the results section, this parts of the architecture extract useful features for the next stages, where the final lesion segmentation is performed.

At the second stage, features from the first stage are combined using a statistical model and combine into a single rich representation of the data that is aware of local, global and context properties for each voxel from the original input. Several approaches were tested and compared at this processing stage, in which the joint distribution by 4 different overall architectures, including Artificial Neural Networks (ANNs), Random Forests (RFs) and Convolutional Neural Networks (CNNs). This stage will be shown to yield better classification performance than the first stage, and also to obtain a set of feature maps, from which one could learn more about relevant structures in the brain by reverse-engineering.

MS lesion classification over MRI varies over different schools. While this renders the challenge more difficult, it also opens the opportunity to reveal some of the hidden ingredients that lead to the ground-truth labeling. Deep learning approach will be advantageous in terms of:

- Visualization of these hidden considerations;
- Examination of their consistency between patients;
- Quantification of their discriminative power;

In this thesis, I will present a large amount of deep learning experiments over the clinical trials data, while benchmarking their performance, and presenting qualitative results. The results will be comparable to existing lesion segmentation methods that were trained over the same dataset. As mentioned above, obtaining comparable scores would be interesting not only in terms of classification, but also for further exploring the deep-learned features. The contributions of this research are described in the next section.

1.3 Contributions

The work presented in this thesis claims to contribute to the field of lesion detection in magnetic resonance imaging as follows:

1. ***A general deep learning framework for the detection and segmentation of MS lesions in brain MRI***. Most lesion segmentation frameworks to date include hand-crafted features. The proposed architecture uses the CNN framework where features extraction is automatically inferred from the system. Although these types of methods are popular in computer vision, their integration into pathology detection and segmentation tasks in the context of medical imaging is still wide open. The proposed architecture was validated for robustness over two large clinical trials, containing 1,175 different patients, and acquired in multiple sites worldwide using different scanners. This proposed method could be generalized and used for other brain lesion segmentation tasks, such as stroke lesions segmentation or brain tumor segmentation.

2. ***Automatic inference of CNN brain-features in the presence of pathology.***. These features were automatically inferred from a large amount of data in the presence of pathology, and could be applied to different medical image analysis context. The experiments show intra- and inter-patient consistency over these features and could replace traditional atlases in the context of medical image segmentation.

1.4 Thesis outline

This thesis presents a deep learning approach for MS lesion segmentation using convolutional neural networks.

Chapter 2 presents the literature review, discusses the notion of an MS lesion within the MRI domain, and introduces the reader to fully automated techniques in the context of MS

lesion segmentation. The challenge along with commonly used components for the lesion segmentation pipeline are explained, and related work is summarized and compared. I present an overview on the different approaches used in the literature to address the task, their main differences, such as classification, clustering and outlier-detection, along with their validity that could be inferred from the extent of the data on which the techniques were validated. This chapter will continue with a literature review of convolutional neural networks, showing applications in the computer vision domain, along with their recent results in the field of object recognition. The review continues to medical imaging application of CNNs, presenting several recent publications along with their performance and comparison to previously available methods. Lastly, several recent papers in which pathology classification using deep learning techniques are presented and discussed. This part of the chapter motivates the usage of CNNs for MS lesion segmentation by presenting very similar tasks for which CNN-based methods were successfully applied.

Chapter 3 presents the theory behind artificial neural networks (ANN), activation functions, multi-layer perceptron (MLP), convolutional neural networks (CNN), capacity, back-propagation and prediction. It discusses the advantages of CNN for computer vision tasks from the theoretical point of view, and explains different hyperparameters used for the training.

Chapter 4 presents the proposed architectures, the reasoning behind their design, and the models used for each component. The design considerations are introduced, along with the expectations from each of the different components, while taking the effects of dataset quality into account. Three different architectures were proposed, each comprised of 4-5 models as its building blocks.

Chapter 5 contains the experimental results. In this chapter, the framework is applied to 2 large, multi-center, multi-scanner clinical trials, comprising a total of 1,175 patients. First, the method was configured and tested over the first clinical trial only. Once the hyper-parameters were set, no further tuning was allowed and the architecture was tested over the second clinical trial, which is much larger, and showed similar performance. The results of the method over this data yielded sensitivity values of up to 0.68, and Dice scores up to 0.59 when tested over both of the large trials. The method achieved even higher metric scores of 0.86-1.00 true-positive rates when considering only larger lesions. The results from more than 30 different experimental configurations will be compared. The design choices behind the configuration will be exposed, starting from the sampling of the input data. Sampling is an important element for the technique, and therefore the considerations are presented and

discussed in this chapter. The evaluation metrics used for obtaining the qualitative results are described in detail, including voxel-connectivity models and lesion-wise metrics. The chapter then continues to present the results from each of the classifiers, both qualitatively and quantitatively, while exposing a large amount of results over many experimental architectures for each component. This chapter includes a discussion regarding the inclusion of atlas priors in the classification scheme. Lastly, the chapter presents the hidden features extracted by the convolutional neural networks, and examines their discriminative power, consistency and general appearance, along with their results over an additional dataset.

Chapter 6 presents the conclusions, and explains the major contributions of this thesis. It contains a discussion about the experimental results of the proposed methods over the clinical trials data, and connects them to the original design considerations. It also discusses the features that were automatically inferred by the convolutional neural networks and their inter-patient consistency. Lastly, ideas and improvements for future research are presented, while suggesting additional tasks for which the method could be applied.

Chapter 2

Background and Literature Review

In this chapter, I will discuss the topic of automatic Multiple Sclerosis lesion segmentation, presenting the main challenges and characterization of MS lesions. I will continue by presenting previous approaches for the task, which are divided into two categories of methods: (a) supervised, and (b) unsupervised. The common flow of lesion segmentation techniques will be discussed, followed by an extensive literature review of published MS lesion segmentation methods that will motivate the need for convolutional neural networks. The last section will present a literature review of recent convolutional neural networks applications, starting at the computer vision domain and continuing to medical image analysis and pathology detection. The purpose of this chapter is to motivate the reader and prepare the ground for the proposed architecture that applies deep learning to the challenging task of MS lesion segmentation.

2.1 Automatic Lesion Segmentation

At the time of writing this thesis, a widely consistent set of manually designed features that define white-matter Multiple Sclerosis lesions in magnetic resonance imaging does not exist [29]. The development of supervised automatic methods for MS lesions classification and segmentation is mostly dependent on labeled data examples, rather than applying an algorithm that correlates them to pre-defined criteria. Guidelines suggest that MS lesions are in general brighter than their surroundings within the white matter when reading proton density (PD) or T2-weighted images, however, in fluid-attenuated inversion recovery (FLAIR) images these lesions are either brighter or darker, depending on their nature and severity. In terms of location, lesions are usually found centered about small blood vessels, and frequently

occur in the juxtacortical and infratentorial regions [29] [26]. The spatial distribution of MS lesions, modeled by Rohit et al. [11] over a database of 84 subjects is shown in Table 2.1. Different Multiple Sclerosis lesions are shown in Figure 2.1.

In general, MS lesions are categorized into 3 major classes: (a) T2w lesions, (b) Gadolinium enhanced lesions, and (c) black holes. Comparing to white matter, T2w lesions may be either hyper or isointense in T1w, and hyperintense in FLAIR, PD- and T2-weighted images. Gadolinium enhanced lesions increase their T1w intensities after the patient was injected with the contrast agent, and observed as hyperintensities in FLAIR, PD- and T2-weighted images. Black holes are hypointensities in T1w-weighted images that do not enhance after a gadolinium injection, they appear as hyperintensities in FLAIR, PD- and T2-weighted images. [29]

Full automatic lesion segmentation techniques often share a common processing structure, which is generally referred to as the segmentation pipeline [29]. The pipeline is comprised of several stages, beginning with the original images as inputs and ending with the segmentation masks. First, the images are spatially aligned to the same image space, such as Talairach [20]. This process is named registration, and employs methods that are mainly landmark-, segmentation- or voxel-based, where landmarks could be either anatomical or geometrical, segmentation could be using rigid models, such as surfaces, curves and points, or deformable models like snakes or nets [59]. The second stage is the skull-stripping, or brain extraction, in which the brain is delineated and a brain-mask is generated, providing the next stages with the ability to strictly process the brain region. The next stage corrects the effect of spatial inhomogeneity of voxel-intensities, caused by the inhomogeneity of the magnetic fields of the scanner used for the image acquisition. The fourth stage is an overall reduction in the noise over the image, in which spatial information is generally used. The last stage before the segmentation itself is an inter-patient normalization of the voxel intensities, resulting in similarly appearing images, even though acquired with different parameters and/or scanners in different sites. The pipeline is shown in Figure 2.2. Although segmentation methods do not all share the exact same pipeline, the stages described above are used as modular building blocks for their distinct processing. It is important to keep in mind that the sequence of these building blocks is only a general guideline, and frequently altered and adapted to the segmentation technique at hand by reordering, removal or addition of new types of building blocks. [29]

Knowing the common components, we are now ready to continue and explore different MS segmentation methods and their details through the literature review presented in the

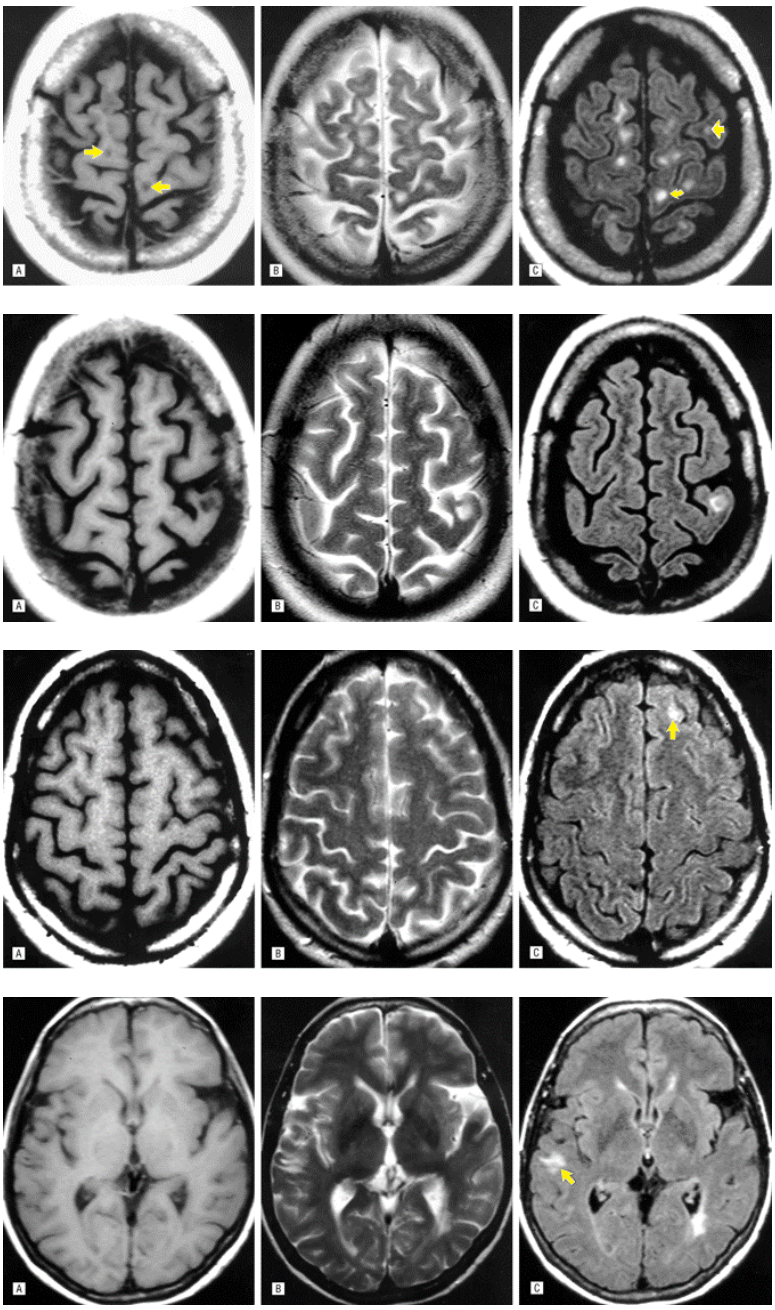


Figure 2.1 Multiple Sclerosis Lesions, courtesy of [11]. Each row is a different patient. Left to right: T1-weighted, T2-weighted and FLAIR images. In the first row we observe MS lesions in the FLAIR image, however, not all are visible in the T2-weighted and the some (arrows on A) appear darker in the T1 image. In the second row we can observe a large 12-millimeter cortical-subcortical lesion over all three modalities. In the third row we can hardly observe the 8-millimeter lesion (marked with an arrow on C) on the T1 and T2 images, but could clearly detect it on the FLAIR image. In the last row, the marked lesion (arrow on C), that extends from the superior temporal cortex to the subcortical white matter is clearly seen in the FLAIR, but not using the T1 or T2 modalities.

Frequency and Distribution of Lesions on FLAIR Scans in 84 Patients With Multiple Sclerosis*

Location	Total No. of Lesions	Mean No. of Lesions per Patient	Total Cortical Lesions, %	Total Brain Lesions, %
Superior frontal cortical	537	6.4	66	17
Inferior frontal cortical	29	0.4	4	0.9
Superior parietal cortical	96	1.1	12	3
Inferior parietal cortical	10	0.1	1	0.3
Temporal cortical	95	1.1	12	3
Occipital cortical	43	0.5	5	1
Total cortical lesions	810	9.6	100	26
Total brain lesions	3072	36.5	. . .	100

*FLAIR indicates fluid-attenuated inversion recovery magnetic resonance imaging with fast spin-echo (see the "Subjects and Methods" section); cortical, located in cortical gray matter (see the "Subjects and Methods" section).

Table 2.1 Spatial Distribution of Multiple Sclerosis Lesions, courtesy of [11].

following section. First, we will explore the types of lesion segmentation techniques and discuss the conceptual differences between them. Second, we will review related work in the field of MS lesion segmentation and motivate the need for context-aware methods. Following, we will present a literature review of Convolutional Neural Networks (CNNs) and their recent applications in order to motivate the concept of automatically extracted features, which will later be used by the proposed method of this thesis. The main objective is to provide the reader with background regarding the two disciplines that meet through this research: (a) medical imaging, and (b) machine learning.

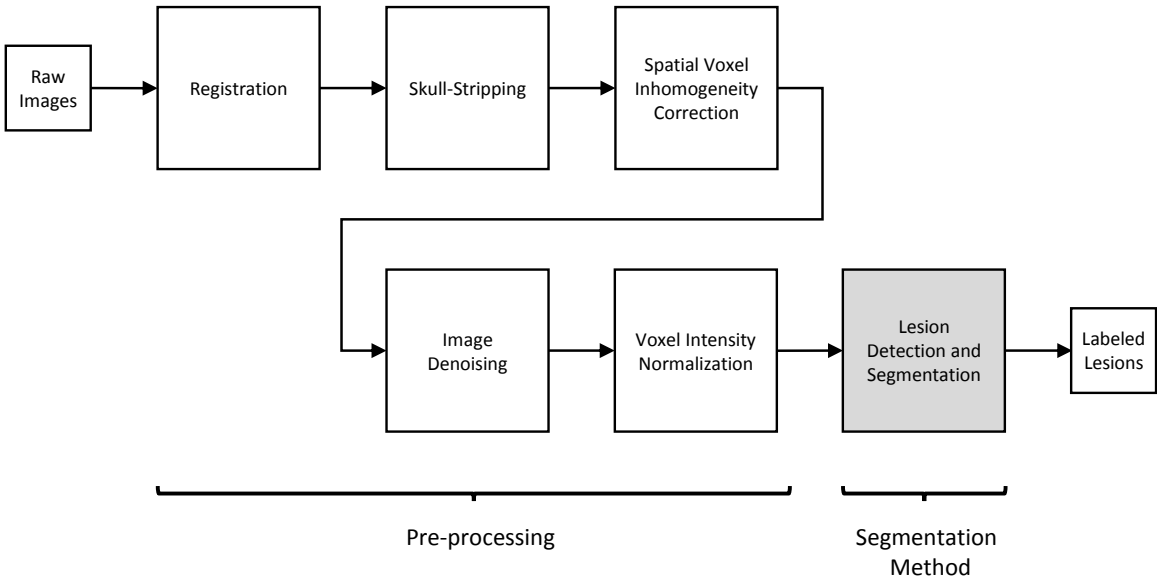


Figure 2.2 Image Segmentation Pipeline. The raw input images typically pass through these stages to obtain the final lesion labels, however, the presented sequence of building blocks is only a general guideline, and frequently adapted to the segmentation technique at hand.

2.2 Related Work on MS Lesion Segmentation

Multiple Sclerosis lesion classification could be divided to two main groups of methods: supervised and unsupervised. Unsupervised approaches are based on a known *a priori* methodology used by labeling experts, which is then simplified and modeled as an algorithm that operates on the input images [29]. Unfortunately, the assumptions that are made are often subjective and require extensive verification if they are to be used in real clinical practice. On the other hand, supervised machine learning methods learn the model and the task from available data. In the case of MS lesions, lesion labels are provided manually. A large number of reliably labeled databases are not publicly available, and several of the following papers have used private images or proprietary clinical trials instead, a fact that leads to difficulty comparing competing methods. [29]

The heart of most supervised methods relies in the process of feature-selection for the classification task, followed by the training of a classifier over their values for a given set of labeled examples. Some methods use the voxels intensities directly, and therefore we could generalize and consider these intensities as the actual features used by the classifier.

2.2.1 Supervised Methods

In 1995, Kamber et al. [45] suggested employing a model for healthy brain tissue probabilities, and by doing so reduced the rate of false-positive MS lesion classification by 50-80%. The classifiers used in their model were decision trees, minimum distance [88] and Bayesian [24] over the intensities of the voxels. They concluded that the annotation of brain tissues was useful for the automatic segmentation of the lesions. In 1996, Johnston et al. [44] published a voxel-wise stochastic relaxation method for fully- and semi-supervised lesion segmentation. They reported that the task of segmenting small MS lesions was challenging, and addressed it by adding inhomogeneity corrections. In 1999, Vinitski et al. [90] implemented a supervised k-NN algorithm to classify tissues into approximately 10 types, including white matter, gray matter, deep gray matter nuclei, cerebrospinal fluid, blood, cyst and subgroups of Multiple Sclerosis plaque. Warfield et al. [92] suggested an algorithm that uses healthy subject templates, often referred to as atlases, then performs a feedback loop between k-NN classification and nonlinear registration, named Adaptive Template Moderated Spatially Varying Classification (ATM SVC). The authors reported that their improved localization of grey matter, by using elastic template registration, allowed them to further penalize false-positive detections and obtain better discrimination between MS lesions and gray matter. The authors also mentioned that their algorithm may fail to distinguish between voxels for which all channels, intensities and spatial priors, share similar intensities. Neither the dataset, the amount of subjects, the resolution or the strength of the magnetic field used by the MR scanner were described in this paper. In 2006, Wu et al. [94] published a paper about automatic classification of MS lesions into 3 types: (a) T2 hyperintense lesions, (b) T1 black holes, and (c) enhancing lesions. They used a k-NN classifier, combined with template-driven segmentation of brain tissues and lesion types. The method employed inhomogeneity correction, followed by a semi-automatic intracranial cavity masking and an additional inhomogenity corection only over the brain-masked portion. The classifier used for the task was statistical intensity-based k-nearest neighbours, followed by template-driven segmentation and partial volume artifact correction. The authors reported that T1 black holes achieved the lowest segmentation sensitivity, while Gadolinium-enhancing lesions were detected with the best sensitivity. At the same year, Harmouche et al. [36] presented a fully automatic method, based on a Bayesian approach, for segmenting lesions and normal tissues. They used a Markov Random Field (MRF) in order to assure local smoothness, and introduced a probabilistic measure for the classification confidence. The authors reported significant improvement in the results when spatial information was integrated into the feature-set used for the segmentation. As for the

preprocessing, the authors applied bias field correction, extraction of brain parenchyma and intensity normalization.

Morra et al. [60] proposed a framework that learned a unified appearance and context model from the data. They used a pool of more than 18,000 features for describing position, intensity, tissue prior and neighborhood-wise properties, such as standard deviation, mean, curvature, gradients and wavelet decomposition coefficients. The data was downsampled and bias field corrected in the preprocessing stage, and the features were classified using AdaBoost with decision stumps over the image and probability maps. Wels et al. [93] published a fully automated approach that is based on probabilistic boosting trees over an overcomplete set of Haar-like features. Their technique learned a set of discriminative features for obtaining posterior probabilities using the ensemble boosting, then the results were refined using a standard level set segmentation after applying a stochastic relaxation stage, while modeling the data in a Markov Random Field. Kroon et al. [49] implemented a method that uses a Principle Component Analysis (PCA) model, feature vectors and atlas registration for the MS lesion segmentation challenge 2008. The features used in their model are voxel intensities, neighbouring voxels, local histograms, normalized location and atlas based prior probabilities. For preprocessing, they proposed a genetic bias field correction in addition to an atlas based correction and edge preserving filtering. However, they reported that their bias correction methods introduced artifacts that lead to false-positive detections, and therefore the model performed better using the original images. Scully et al. [72] published a non-parametric approach that uses KMeans segmentation followed by a Naive Bayes classification for the challenge. Their model learned the joint intensity histograms over FLAIR, T1- and T2-weighted images for each tissue class, along with neighbouring voxels and location information. The authors reported that their approach could have been improved by additional intensity standardization, or by adding a Markov Random Field over the class labels to incorporate spatial information. The authors also indicated that there was a possibility that the labeling of the dataset was inconsistent and inaccurate, along with a number of artifacts in the scans such as motion and noise. In 2009, Subbanna et al. [80] presented a fully automated MRF-based method that embedded neighbourhood information along with local variations into their tissue model. Their method classifies voxels into six classes: (a) background, (b) white matter, (c) grey matter, (d) cerebrospinal fluid, (e) T1-hypointense lesions, and (f) T2-hyperintense lesions. The healthy tissues and lesions were modeled as multivariate Gaussians over the intensity channels, and the method was applied after bias-field correction, brain extraction and intensity normalization as preprocessing steps.

The authors suggested that their results could have been improved by incorporating different models for different regions in the brain, or by fully integrating the classification scheme and smoothing function to obtain less lesion over-estimations. Akselrod-Ballin et al. [8] proposed a method that detects abnormal brain structures, using a multiscale approach, by combining classification and segmentation. Their method performs hierarchical decomposition of the MR images in order to produce features that describe the location, shape, intensity and neighbourhood of the voxel, followed by a decision forest tissue classification. The authors concluded that their experiments could have been improved by using higher resolution of 3mm instead of 5mm for the thickness of the slices.

Yamamoto et al. [96] suggested a scheme for reducing false positive detections, which relies on rule-based, level set method and Support Vector Machine (SVM). Their method used FLAIR, T1-, and T2-weighted MR images and enhanced them, based on background subtraction, followed by initial identification of lesion candidates using linear discriminate analysis over the T1 voxel intensity. The candidates were selected according to a set of heuristics described in the paper, then segmented using a region-growing technique based on their geometry. Their next stage was the extraction of gray-level intensity features used to train a support vector machine, followed by reduction of false positive outliers using a rule-based and a level set method. A year after, Geremia et al. [32] published a paper about automatic lesion segmentation using a discriminative random decision forest using FLAIR, T1- , and T2-weighted MR images along with spatial priors. They introduced context-rich features that compare the voxel of interest to distant regions, measuring symmetry and other neighbourhoods. The authors reported that the normalization was an important preprocessing step for obtaining their results, and used the MICCAI 2008 challenge database for evaluation.

The supervised techniques use features in order to classify MS lesion. These features are Gabor decompositions in Subbanna et al. [80], location/shape/intensity in Akselrod-Ballin et al. [8], context-rich features in Geremia et al. [32] and more. These features are handcrafted and manually selected.

We will now continue reviewing the unsupervised methods, which are mostly based on outlier detection.

2.2.2 Unsupervised Methods

In 2001, Van Leemput et al. [87] published a fully automatic unsupervised atlas-based method for lesion segmentation. The method detected MS lesions as outliers, while modeling

healthy brain tissues using a Markov Random Field. The method is an iterative Expectation-Maximization (EM) algorithm that classifies data into healthy tissue classes, followed by outlier detection based on neighbouring voxels, and repeated until convergence. Anbeek et al. [10] proposed a fully automated method, based on a k-NN classifier over the T1-weighted and FLAIR modalities. Their method used both spatial information and voxel intensities as features and provided probabilistic segmented images, which are then passed through a certain threshold value to obtain the final results. The authors generated brain masks at their preprocessing stage, and reported that tissues like skull and skin disturbed their classifier. They also suggested that having a probabilistic output for tissue classes was advantageous due to the flexibility it could provide to further processing stages. Souplet et al. [78] used the FLAIR, T1-, and T2-weighted modalities for the MICCAI 2008 MS lesion segmentation challenge. Their method used the T1w and T2w images to segment the brain into compartments, followed by intensity thresholding over the FLAIR image. This method required intensity normalization at the preprocessing stage, that allows using an EM algorithm over the intensities to segment tissues into 10 different gaussian-modeled classes, where outliers according to a certain Mahalanobis distance threshold are considered as lesions. In 2010, Shiee et al. [74] developed a method for lesion delineation using a topological and statistical atlases. The method is a generalization of their healthy tissue segmentation method named Topology-preserving Anatomical Segmentation (TOADS) [12], by considering lesions and other topological outliers as topology-preserving when grouped with underlying tissues [74]. Their method addresses lesions by adding an additional class to the model and adjusting the tissue weights accordingly. The authors suggested that another class should have been added in order to accommodate the two lesion subtypes: T1 black holes and T2 hyperintense. Garcia-Lorenzo et al. [30] developed a method using normal appearing tissue intensities and a trimmed likelihood estimator . The authors preprocess the data for intensity inhomogeneties and skull-stripping using the T1-weighted image, followed by estimation of the trimmed likelihood model for normal appearing brain tissues. The outliers from the previous stage, based on the Mahalanobis distance, are then marked as lesion candidates that are eliminated using a priori heuristic rules based on intensity, size and neighbour information. The authors concluded that outliers included lesions, however, other kinds of voxels such as vessels, skull tissues or acquisition artifacts were considered outliers as well and labeled during the outlier detection process.

2.2.3 Neural Network Based Methods

The last category to be discussed is methods that employ artificial neural networks, which belong to the category of supervised techniques. This techniques are presented separately due to their relevance to the topic of the research presented in this thesis.

In 1994, Zijdenbos et al. [98] developed a semi-automatic approach for lesion segmentation based on an artificial neural network classifier, while adding noise filtering at the preprocessing stage and surface-fitting method for correcting the variations of spatial intensity. They reported the cardinality of the pre- and post-processing routines, like heuristically eliminating lesions measuring less than 8 square millimeters, to achieve comparable results to the inter- and intra- rater of manual expert segmentations. Goldberg-Zimring et al. [33] developed an automatic method for delineation of MS lesions that is based on detection of voxel hyperintensities, followed by removal of false-positives by features such as anatomical location, shape and size using an artificial neural network classifier. The reported that the shape-index feature used to quantify the shape of lesion candidates was useful for the task, and was defined as the ratio between the area and the squared perimeter of the object. On the other hand, the authors noted a known limitation of the algorithm for cases in which there is poor contrast between lesions and other brain tissues. In 2002, Zijdenbos et al. [99] proposed an automatic pipeline, including intensity normalization, noise reduction, intensity nonuniformity correction, registration, resampling, and brain-masking for preprocessing, followed by artificial neural network based tissue classification. Hadjiprocopis et al. [35] developed a method that uses an ensemble of artificial neural networks over proton density and T2-weighted magnetic resonance images, while each neural network was trained on a different subset of the multiple sclerosis subjects training data. The authors suggested that their results could have been significantly improved by adding a post-processing stage with a priori knowledge about the brain anatomy. In 2007, Younis et al. [97] applied an artificial neural network based approach, using BrainWeb [18] simulated T1- and T2-weighted MR images as inputs. They used noise filtering for preprocessing, followed by a preliminary T1 image segmentation into classes of white matter, gray matter, cerebrospinal fluid and Multiple Sclerosis lesions, while incorporating intensity information from the six nearest neighbours for each voxel. Once the image was segmented over the T1 modality, the T2 image is segmented using an additional artificial neural network that operated only on the non-CSF classes from the previous stage, in order to not confuse them with lesions. Worth mentioning is that the authors used a only a single simulated image for training, and 5 simulated images for testing. In 2012, Cerasa et al. [16] presented a cellular neural network, which is based on

genetic algorithms, for automatic lesion segmentation over FLAIR images. Their approach takes spatial interaction between neighbouring voxels into account during the segmentation process, and satisfactory results were reported by the authors, however, the research was conducted over a dataset of only 11 patients, in which very poor results were reported for one of the testing subjects and were attributed to the MR image intensity range. Also, the approach was applied on a single modality, rather than over a multichannel MRI.

The methods above use different kinds of neural networks as their classifiers. However, none of them uses convolutional neural network, which is an architecture that recently gained popularity in the field of computer vision for object classification tasks. The input features used for the methods above were manually selected, and an advantage that convolutional neural networks could provide would be to make this process automatic, and therefore less subjective.

The next section presents a literature review containing applications of convolutional neural networks in recent literature. The review will start by presenting applications from the field of computer vision and object recognition and will continue with their applications in the field of medical image analysis and pathology detection. As will be shown, these architectures have recently gained popularity mostly due to their unbeatable performance on large-scale object recognition tasks such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [68].

Method	Classifier/Clustering	Subjects	Images	Magnetic Field	Voxel Size [mm]			Modalities Used								
					X	Y	Z	PDw	FLAIR	T1w	T1c	T2w	MT	FF	DTI-MD	DTI-FA
Zijdenbos et al.(1994) [98]	ANN	6	36	1.5T	0.91	0.91	3./5	✓	✗	✓	✗	✓	✗	✗	✗	✗
Kamber et al.(1995) [45]	DT[1], Statistical Classifiers[2]	12	12	1.5T	-	-	2	✓	✗	✓	✗	✓	✗	✗	✗	✗
Johnston et al.(1996) [44]	MRF	5	5	1.5T	0.78	0.78	5	✓	✗	✗	✗	✓	✗	✗	✗	✗
Goldberg-Zimring et al.(1998) [33]	ANN	14	45	-	-	-	-	✗	✓	✗	✓	✓	✗	✗	✗	✗
Vinitski et al.(1999) [90]	k-NN	12	12	1.5T	-	-	3	✓	✗	✓	✓	✓	✓	✗	✗	✗
Warfield et al.(2000) [92]	k-NN + Nonlinear Registration Feedback	-	-	-	-	-	-	✓	✗	✗	✓	✓	✗	✓	✗	✗
Van Leemput et al.(2001) [87]	Expectation Maximization	50	300	1.5T	0.9	0.9	2.4/5	✓	✗	✓	✗	✓	✗	✗	✗	✗
Zijdenbos et al.(2002) [99]	ANN	600	1000+	-	-	-	3	✓	✗	✓	✗	✓	✗	✗	✗	✗
Hadjiprocopis et al.(2003) [35]	ANN	20		1.5T	0.93	0.93	3	✓	✗	✗	✗	✓	✗	✗	✗	✗
Wu et al.(2006) [94]	k-NN	6	12	1.5T	0.97	0.97	3	✓	✗	✗	✓	✓	✗	✗	✗	✗
Harmouche et al.(2006) [36]	GMM + MRF	10	10	-	-	-	-	✓	✗	✓	✗	✓	✗	✗	✗	✗
Younis et al.(2007) [97]	ANN	1*	5*	-	-	-	1	✗	✗	✓	✗	✓	✗	✗	✗	✗
Anbeek et al.(2008) [10]	k-NN	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✗	✗	✗	✗	✗
Morra et al.(2008) [60]	AdaBoost	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✓	✗	✗	✓	✓
Wels et al.(2008) [93]	PBT[3] + MRF	6	6	-	-	-	-	✗	✓	✓	✓	✗	✗	✗	✗	✗
Kroon et al.(2008) [49]	PCA	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✓	✗	✗	✗	✗
Scully et al.(2008) [72]	KMeans + Nave Bayes	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✓	✗	✗	✗	✗
Souplet et al.(2008) [78]	GMM + EM	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✓	✗	✗	✗	✗
Subbanna et al.(2009) [80]	GMM + MRF	24	24	1.5T	-	-	3	✓	✓	✓	✗	✓	✗	✗	✗	✗
Akselrod-Ballin et al.(2009) [8]	Decision Forest	25	25	1.5T	.83/.98	.83/.98	5	✓	✓	✓	✗	✓	✗	✗	✗	✗
Shiee et al.(2010) [74]	Fuzzy C-Means Clustering	10	10	3T	0.83	0.83	2.2	✗	✓	✓	✗	✗	✗	✗	✗	✗
Yamamoto et al.(2010) [96]	SVM + LSM[4]	3	6	3T	1	1	5	✗	✓	✓	✗	✓	✗	✗	✗	✗
Geremia et al.(2010) [32]	Random Forest	45	45	3T	0.5	0.5	1	✗	✓	✓	✗	✓	✗	✗	✗	✗
Garcia-Lorenzo et al.(2011) [30]	TLE[5]	10	10	1.5T	0.97	0.97	3	✓	✗	✓	✗	✓	✗	✗	✗	✗
Cerasa et al.(2012) [16]	Cellular Neural Network	11	11	1.5T	0.94	0.94	5	✗	✓	✗	✗	✗	✗	✗	✗	✗

Table 2.2 Multiple Sclerosis Lesions Segmentation Methods. Based on the methods presented in [29].

[1]Decision Tree
[2]Minimum distance[88] and Bayesian[24]
[3]Probabilistic Boosting Trees
[4]Level Set Method
[5]Trimmed Likelihood Estimator

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) were introduced by LeCun et. al [52] in 1989, providing an approach to address the challenge of handwritten digits recognition. Even though based on the standard neural network architecture, this new technique has several significant improvements over that make it more suitable for computer vision tasks than traditional ANNs. At 1998, LeCun et al. reached 0.7% [53] misclassification rate over his digits dataset [54], while concurrent CNNs achieved 0.21% [91] error rate, which corresponds to a total of 21 misclassifications over 10,000 test images. The appealing results of convolutional neural networks over this relatively simple classification challenge, and the advances in parallel computation technologies and Graphical Processing Units (GPUs) enabled generating and training considerably larger architectures. In 2014, the ILSVC challenge has introduced a 1,000-classes classification problem over 1,200,000 training images, 50,000 for validation and 100,000 for testing. All the 3 top scoring teams reached error rates between 6.67-7.35% using CNN-based approaches [39, 75, 85]. As of 2015, He et. al. [38] had reported the first solution that surpasses human-level performance (5.1%)[69] for the recognition challenge, achieving 4.94% test error rate.

The major advantage of CNNs lies in the fact that the model learns the optimal set of task-related features from the data, rather than selecting them from a set of predetermined or hand-crafted features. For this reason, CNNs were applied to many challenges from different domains over the several last years. CNNs have been employed for human action recognition [43] in real-world environments, considering time as a third dimension, by using 3D convolutional networks over videos rather than 2D architectures over images. Sermanet et. al. [73] reported 94.85% accuracy and 45.2% improvement comparing to other methods when using convolutional neural networks for the task of house numbers digit classification. Karpathy et. al. [46] used CNNs to achieve a significant improvement, comparing to feature-based baselines, on a large-scale video classification of over 1 million YouTube videos into 487 classes. Sun et al. [82] employed CNNs for facial keypoints detection and reported successfully avoiding spatial local minimas, while being robust to occlusions. These authors reported state-of-the-art results in both detection accuracy and reliability. Toshev and Szegedy [86] used CNNs for human pose estimation, achieving state-of-the-art performance on real-world images. Sun et. al. [83] used CNNs for face representation over 10,000 classes, and achieved 97.45% verification accuracy with weakly aligned faces. Li et. al. [56] employed CNNs for re-identification of pedestrians in disjoint camera views over 1,360 different persons. Samples

from these papers are shown in Figure 2.3.

The applications above address tasks from the fields of computer vision and object recognition. Convolutional neural network is a generic machine learning framework, and have begun to be applied to other domains such as medical image analysis. A literature review of their applications in the medical imaging domain is presented in the following section.

2.4 Convolutional Neural Networks for Medical Image Analysis

In the field of medical image analysis, there were a series of contexts where CNNs were successfully deployed. Convolutional neural networks were used by Curesan et al. [17] for segmentation of neuronal membranes in electron microscopy images, outperforming competing techniques by significant margins. Habibzadeh et al. [34] addressed the challenge of white blood cell differential counts using a CNN, and comparing to the more traditionally used Support Vector Machine (SVM) with Principal Components Analysis (PCA) dimensionality reduction, the authors concluded that the experimental results show that the CNN is more accurate, even in the presence of poor quality samples. In 2013, Prasoon et al. [66] used CNNs for knee cartilage segmentation in MRI scans and achieved an accuracy rate of 99.93%, sensitivity 81.92% and specificity 99.97%. Li et al. [55] published a CNN-based method for completing a missing modality based on available images. The implemented a 3D convolutional architecture for predicting missing Positron Emission Tomography (PET) data from MRI input images, and reported notably outperforming other methods. In 2014, Xu et al. [95] reported that previously developed features like SIFT and Haar were unable to comprehensively represent objects like cells, which are characterized by significant clinical features, and were outperformed by learned-from-data features for the task of cancer cell detection in histopathology images.

Convolutional neural networks were used in 2006 by Ge et al. [31] for computer-aided detection of microcalcifications lesions in mammography. The authors reported cluster-based sensitivity rates of 70, 80 and 90% at respective rates of 0.21, 0.61 and 1.49 false-positives per image. Roth et al. [67] addressed the task of lymph node detection using random sets of deep convolutional neural network observations, achieving sensitivity of 83% at 3 false-positives per volume rate. Cruz-Roa et al. [22] used a deep learning architecture in 2013 for automated basal-cell carcinoma cancer detection. They evaluated different image representations strategies such as bag of features, discrete cosine transform and Haar wavelet transform to learning the features using convolutional auto-encoders and obtaining from-data repre-

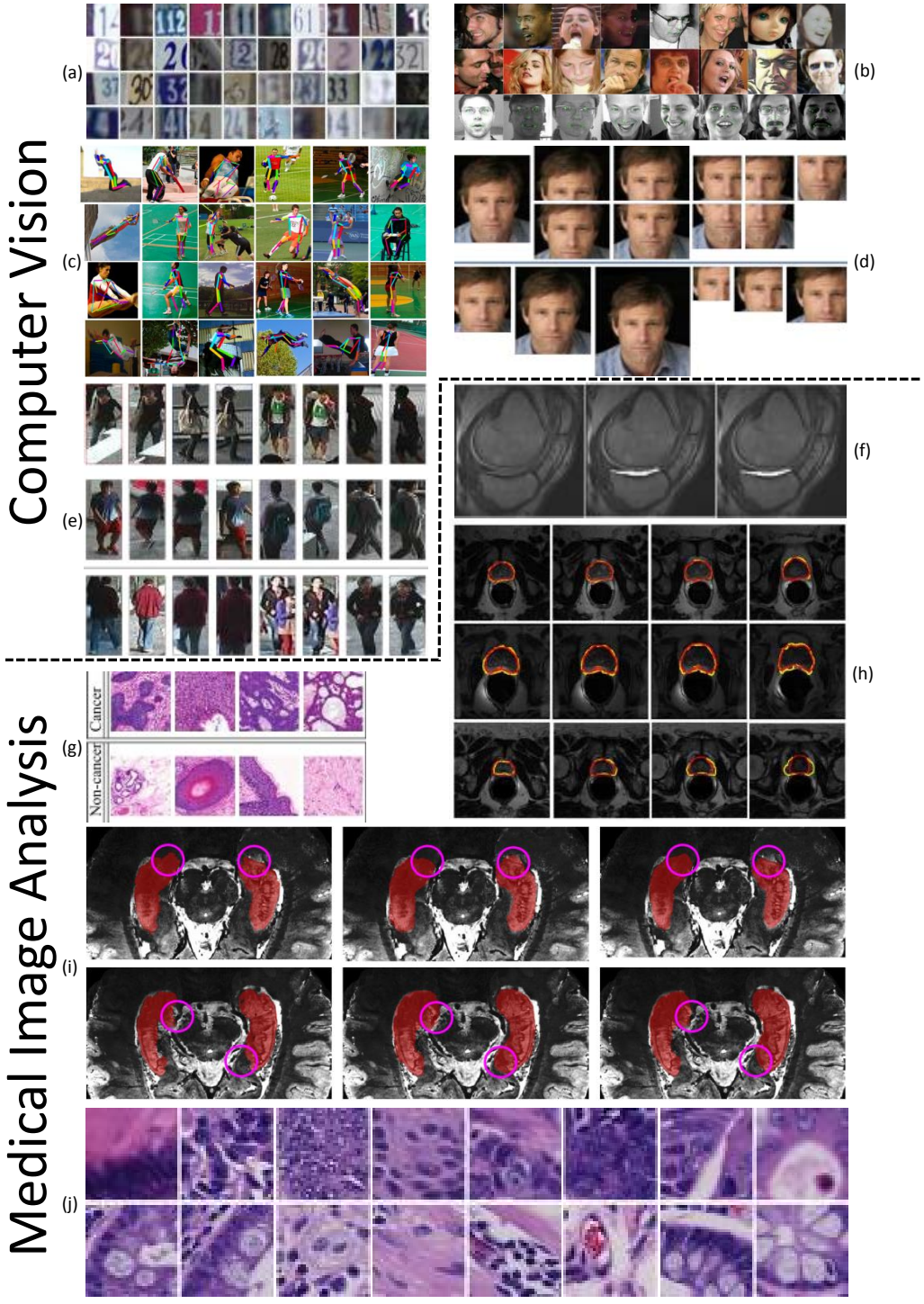


Figure 2.3 Examples of Application Contexts of Convolutional Neural Networks, images are courtesy of [22, 47, 56, 57, 66, 73, 82, 83, 86, 95]. Top to bottom, left to right: (a) house number dataset and classification [61], (b) facial point detection and labels, (c) body pose, (d) face representation, (e) person re-identification dataset and classification, (f) knee MRI and labeling, (g) cancer histopathology images and labels, (h) prostate images and segmentation labels, (i) hippocampus images and segmentation labels, and (j) colon images and cancer classification.

sentations. They concluded that features learned from data had the best task performance of 89.4% in F-measure and 91.3% in balanced accuracy, improving 3% and 7% over bag of features and canonical representations, respectively. Other unsupervised deep learning based approaches were used in recent years for medical imaging. Liao et al. [57] shown that the most effective features for the task of automatic prostate segmentation in MRI were designed in a learning-based manner, rather than hand-crafted without the guidance of the dataset, such as histogram of oriented gradients and Haar wavelets. They used stacked Independent Subspace Analysis (ISA) [51], which applies very similar concepts to the ones employed in convolutional neural networks, achieving state-of-the-art segmentation results. Kim et al. [47] used a two-layer stacked convolutional ISA network and reported promising hippocampus segmentation results, integrating it with multi-atlases based framework and replacing previous hand-crafted features. Lio et al. [58] shown an accuracy improvement when using stacked auto-encoders for computer-aided diagnosis of Alzheimers disease. Brosch et al. [15] used deep belief networks to learn the manifold of brain MRIs, rather than predefining a similarity measure. They reported that the manifold coordinates captured variations between images that correlate with clinical and demographic parameters, such as age and gender. Examples from these publications are shown in Figure 2.3. Recently, convolutional neural networks were applied by Vaidya et al. [70] for longitudinal Multiple Sclerosis lesion segmentation, and achieved comparable scores to the inter-rater variability after a post-processing stage that rejects lesions outside the atlas-registered white matter region. It is worth mentioning that these are the challenge results, and not an official publication.

2.5 Motivation for CNN

The previous section discussed convolutional neural networks and their success in computer vision applications. Medical image segmentation deals with a very large dimensionality of inputs. Most of the MR images used for this research, which are from real clinical trials, are in the orders of 3.9 million voxels per each of the five input modalities, resulting in total input dimensionality of 19.7 million voxels per image. Convolutional neural networks could exploit the 3D structure of the input and also allow some degree of invariance. This is performed using their local-connectivity, constrained sharing of parameters and hierarchical pooling layers. These properties, along with their recent success in the computer vision field, and considering the fact that they produce data-driven features rather than require hand-crafted ones as inputs makes them attractive for experimenting with Multiple Sclerosis lesion

segmentation.

The next chapter will introduce the theoretical background of convolutional neural networks in order to provide the reader with the background required for understanding the proposed method and experimental configurations in the following chapters. The concept of artificial neural networks will be explained thoroughly, starting from the neuron model up to the full CNN model, layer definitions and optimization methods. Also, this chapter will discuss the advantages of convolutional neural networks over traditional fully-connected neural networks within the computer vision domain. By the end of this chapter, the reader should be familiar with both the model and the common training and inference procedures.

Chapter 3

Convolutional Neural Networks Overview

Convolutional Neural Networks (CNN) are part of a larger field referred to as *deep learning*. *Deep learning* is a subfield of machine learning, in which multiple levels of representation and abstraction are built based on the data. For example, it presents strategies for representing highly varying functions using only a few parameters through the composition of several non-linearities [14]. These collections of several non-linearities are referred to as *deep architectures*, which are different from *shallow architectures* which contain only few levels of data dependent computational elements [1, 14, 23]. For example, a neural network with a single hidden layer is a shallow architecture, while multi-layer and convolutional neural networks are deep architectures. Other examples of *deep learning* are: deep belief nets [40], deep Boltzmann machines [71], stacked denoising autoencoders [89] and deep recurrent neural networks [64]. Convolutional Neural Networks are a machine learning technique that recently gained popularity in the field of computer vision, mainly for the task of object classification. It is inspired by the biological vision system, and is built upon the artificial neural network model. The objective of the CNN is to classify the input image, while taking its spatial structure into account using mechanisms such as receptive fields from the human visual system, and providing invariance to translations and other vision-related transformations. The main advantage of CNNs over traditional methods is that they learn features from the data instead of using hand-crafted features. In this chapter, artificial neurons are presented, followed by their usage as the building blocks for neural networks models, and ending with the full convolutional neural network model and training procedures.

3.1 Artificial Neural Networks (ANN)

Artificial neurons are the building blocks of artificial neural networks. The neuron unit receives an input in a form of a vector $\mathbf{x}$ (which could be a feature vector or an input signal for a classification task), performs a dot product of the input with a predefined weights vector $\mathbf{w}$ (could represent the features importance), added to a predefined bias b (used to shift the decision boundary from the origin) and passes the result through an activation function g, that could be used to add non-linearity to the output. This process is defined as $h(\mathbf{x})$ in Equation (3.2), where a is the pre-activation function defined in Equation (3.1), before applying g on the result. The input vector $\mathbf{x}$ and the weights vector $\mathbf{w}$ are both of dimension n, while the bias b is a scalar. The artificial neuron model is shown in Figure 3.1. A list of commonly-used neuron activation functions $g(a)$ and their corresponding derivatives $g'(a)$ are shown in Table 3.1 and visualized in Figure 3.2. In terms of capacity, a single neuron could only linearly separate between classes as shown for the case of $\mathbf{w} \in \mathbb{R}^2$ in Figure 3.3. Equations (3.2)-(3.15) are adapted from [50].

$$a(\mathbf{x}) \equiv \mathbf{w}^T\mathbf{x} + b \tag{3.1}$$

$$h(\mathbf{x}) \equiv g(a(\mathbf{x})) = g(\mathbf{w}^T\mathbf{x} + b) \tag{3.2}$$

$$\mathbf{x}, \mathbf{w} \in \mathbb{R}^n, b \in \mathbb{R}^1 \tag{3.3}$$

Activation Function Name	Symbol	Expression	Derivative
Linear		a	1
Sigmoid	$sigm(a)$	$\frac{1}{1+e^{-a}}$	$sigm(a)\,(1 - sigm(a))$
Hyperbolic Tangent	$tanh(a)$	$\frac{e^a - e^{-a}}{e^a + e^{-a}}$	$1 - (tanh(a))^2$
Rectified Linear Unit	$relu(a)$	$max\{0, a\}$	$\begin{cases} 0 \text{ if } x < 0 \\ 1 \text{ if } x > 0 \end{cases}$
Softplus	$softplus(a)$	$log(1 + e^a)$	$sigm(a)$

Table 3.1 Common activation functions for artificial neurons.

A single neuron can only linearly separate data, and therefore data that is not linearly separable could not be correctly classified using such model. This limitation in terms of capacity had led to the development of a multilevel representation technique called Multi-Layer Neural Network, or Multi-Layer Perceptron (MLP).

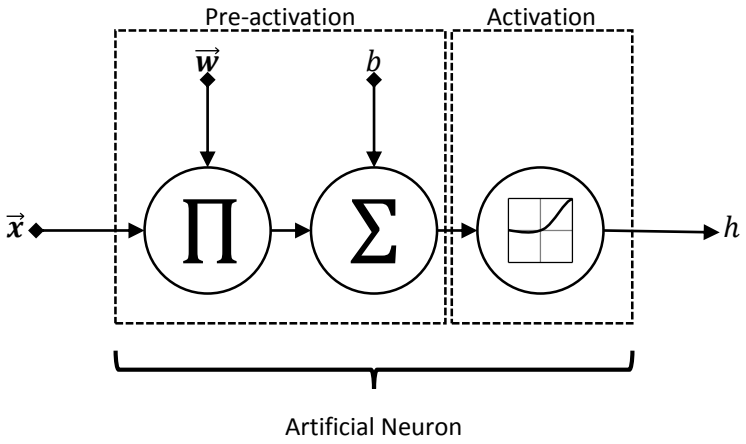
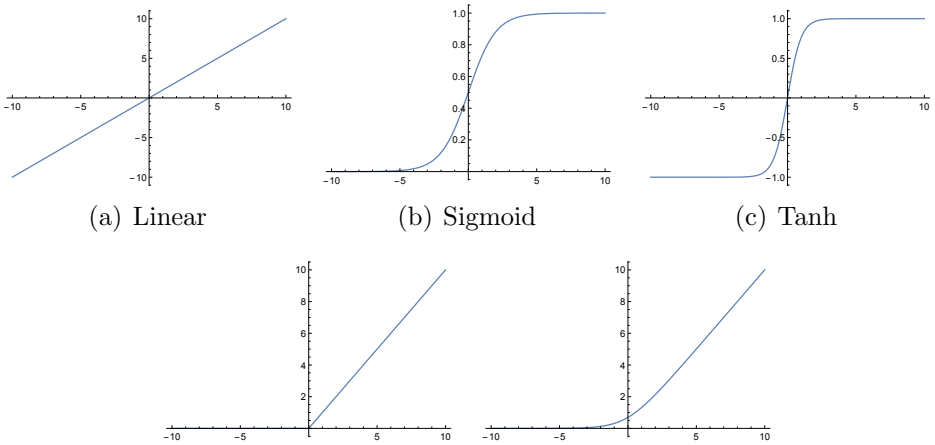


Figure 3.1 Artificial Neuron Model. The input signal x passes through the pre-activation stage on the left hand side of the figure, then continues to the activation function, yielding h as the output signal. The free parameters of the neuron are the wights vector $\mathbf{w}$ and the bias b.



(a) Linear (b) Sigmoid (c) Tanh

(d) ReLU (e) Softplus

Figure 3.2 Common Activation Functions.

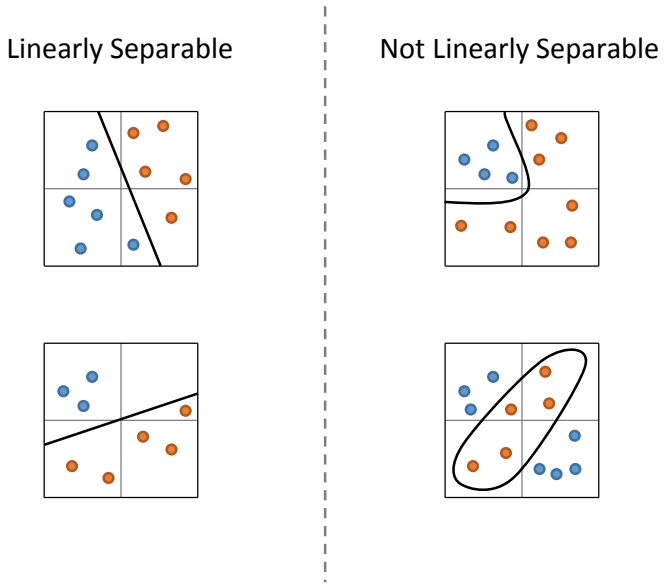


Figure 3.3 The Capacity of a Single Neuron. A single neuron can only linearly separate data. On the left hand side, we show two examples of linearly separable two-dimensional data that a single artificial neuron could classify correctly. On the right hand side are two non linearly separable scenarios for which a single neuron could not correctly classify all the data points.

3.2 Multi-Layer Perceptron (MLP)

MLPs generate a new representation of the input-space that converts the data to a linearly separable set, which for the computer vision task maps to a features vector that was extracted from an input image. This representation is formed by adding an additional layer of neurons between the input and the output as shown in Figure 3.4. This process is defined by the pre-activation function of the k th hidden layer in Equation 3.4. The values of the hidden layer are then calculated by applying the activation over the units in Equation 3.5 for every layer according to 3.7. The output values $\mathbf{y}$ are determined by the output function $\mathbf{o}$, which could be used to constrain the sum of the output neurons to be one in the case of classification, for example. Note that each layer k could contain a different amount of artificial neurons N_k. In terms of capacity, and the universal approximation theorem, a standard MLP with only a single hidden layer is capable of approximating any continuous function to any desired degree of accuracy when providing a sufficient amount of hidden units [41].

$$a^{(k)}(\mathbf{h}^{(k-1)}) = \mathbf{W}^{(k)}\mathbf{h}^{(k-1)} + b^{(k)} \tag{3.4}$$

$$h^{(k)}(\mathbf{h}^{(k-1)}) = g(\mathbf{W}^{(k)}\mathbf{h}^{(k-1)} + b^{(k)}) \tag{3.5}$$

$$\mathbf{y}(\mathbf{x}) = \mathbf{o}(\mathbf{W}^{(H+1)}\mathbf{h}^{(H)} + b^{(H+1)}) \tag{3.6}$$

$$1 \leq k \leq H; h^{(0)} \equiv x \tag{3.7}$$

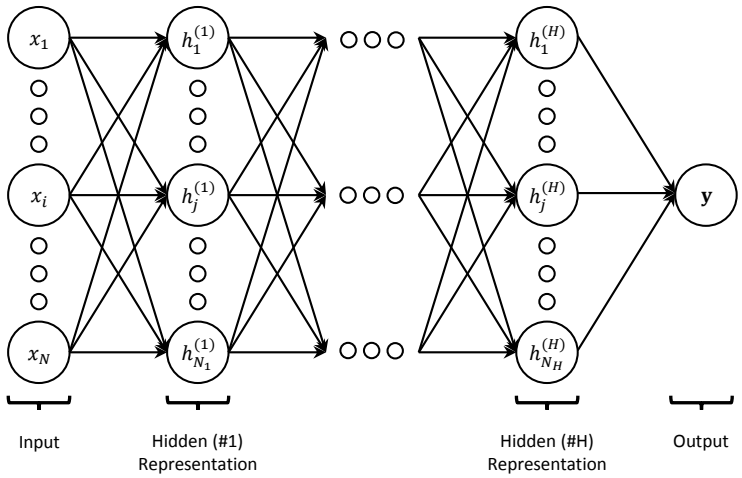


Figure 3.4 Multi-Layer Perceptron. The input data $\mathbf{x} \in \mathbb{R}^N$ is being represented as $\mathbf{h} \in \mathbb{R}^H$ in each of the hidden layers, then classified as $\mathbf{y}$ at the output stage.

The limitation of MLP in the context of computer vision is that they do not model the spatial relationships between the pixels, which makes them sensitive to image transformations such as translation and scaling. Convolutional Neural Networks (CNNs) address these issues by using three mechanisms that provide invariance to such transformations.

3.3 Convolutional Neural Networks (CNN)

Convolutional neural networks are based on the multilevel perceptron model, yet with several characteristics that make them more appropriate for computer vision related tasks. Vision problems, in which the inputs are images, introduce a very large input dimensionality. For example, using a standard camera to take pictures of sizes between 1280x960 and 3264x2448 pixels will produce input images of 1.2 to 7.9 million dimensions. In order to build invariance and exploit the known image topology, convolutional neural networks introduce the following three main differences for performance improvement:

1. Shared weights;

2. Local connectivity (receptive fields);

3. Pooling layers (image size reduction by sub-sampling);

The inspiration for local connectivity in neural networks comes from Hubel and Wiesel's paper in 1962 [42], in which they published their discovery of local receptive fields in the visual system of cats. The local connectivity enables the network to extract locally meaningful features such as corners, edges, orientations etc. The artificial model that imitates this behavior comes in the form of convolutions of small kernels of weights with the image, which is equivalent to having a fully connected layer that shares the weights between every local set of connection and has zero weights outside of this neighbourhood. The results of this operation are called feature maps, and the biologically-inspired property of receptive fields is obtained by the limited size of the kernels used for the convolution. The network extracts several features at each location, which is defined by the amount of kernels used to generate them. Note that a shift in the input space will only result in a shift of the output, conserving the same values. Moreover, smaller amounts of training data are needed due to the sharing of weights that reduces the capacity of the model. This also makes the model more robust to distortion. The next stage is subsampling, or pooling, in which the spatial dimensionality of the feature maps is significantly reduced. This could be done by many different subsampling techniques, but the most commonly used are either max-pooling or average-pooling. The first technique propagates the highest activation value within the neighbourhood to the next level in the hierarchy, and the second one propagates the average activation value. This operation builds higher-level representations by combining lower level features and increasing the size of the receptive fields [53]. The architecture of a standard convolutional neural network is shown in Figure 3.5.

3.4 Learning the Parameters

Given a model for an artificial neural network, we now seek to learn the optimal set of parameters $\mathbf{W}$ and b for each layer. In order to frame the learning task as an optimization problem we use the definition of empirical risk minimization in Equation 3.8, where $\mathbf{\Theta}$ is the set of all free parameters of all layers, T are the training examples, l is the loss function, f is the output of the neural network for input $x^{(t)}$ given the parameters, $y^{(t)}$ are the training labels and Ω is a regularizer function over the parameters, weighted by the coefficient λ.

$$\mathbf{\Theta} = argmin_{\mathbf{\Theta}} \frac{1}{|T|} \sum_{t \in T} l\left(f(x^{(t)}; \mathbf{\Theta}), y^{(t)}\right) + \lambda \Omega(\mathbf{\Theta}) \tag{3.8}$$

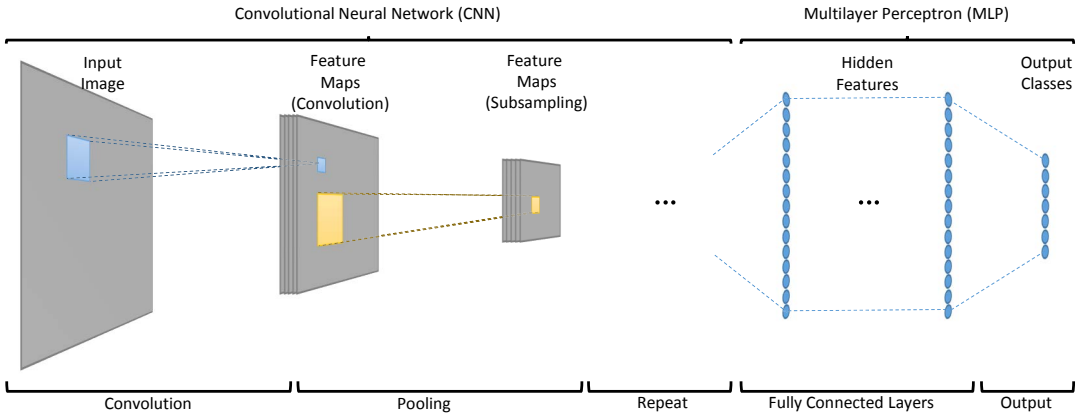


Figure 3.5 Convolutional Neural Network. The input image is convolved with the kernels of the convolutional neural network to obtain the feature maps. These maps are then subsampled at the pooling stage, and the process is repeated, alternating between convolution and pooling. At the end, the results are provided as inputs to a standard multilayer neural network that provides the final classification.

For classification problems, we minimize the negative log-likelihood:

$$l(f(x^{(t)}), y^{(t)}) = -\sum_{c \in C} \delta(y, c) log \left[f(x^{(t)})_c \right] \tag{3.9}$$

The output class is noted as c, and the collection of all possible outputs is noted as C.

$$\delta(x_1, x_2) \equiv \begin{cases} 1 & \text{if } x_1 = x_2 \\ 0 & \text{if } x_1 \neq x_2 \end{cases} \tag{3.10}$$

Where C is the collection of all possible classes, δ is the identity function defined in Equation 3.10 and $f(x^{(t)})_c$ is the notation for taking the cth element of the vector $f(x^{(t)})$. In order to minimize this function, it is common to use the framework of Stochastic Gradient Descent (SGD), and define the term *epoch* as single iteration over the whole training set. The main reason for using the stochastic version of gradient descent rather than the standard one is technical. Large training sets cannot be fully stored in memory at once, especially when using graphical processing units for parallel computation. A tradeoff between gradient descent and stochastic gradient descent is offered by the *batch gradient descent algorithm*. This algorithm is exactly the same as SGD, except that it loops over the whole training set and accumulates the changes over the free parameters Θ. For example, having a dataset of 1000 examples and

input : Labeled training examples $(x^{(t)}, y^{(t)})$ and a loss function $l(x, y; \Theta)$
output: Optimization over Θ

Initialize Θ_0;
for *N iterations (epochs)* **do**
  for *each training example* $t \in T$ **do**
    calculate gradient $\nabla_\Theta l \equiv \nabla_\Theta l(x^{(t)}, y^{(t)}; \Theta)|_{\Theta=\Theta^i}$
    update parameters $\Theta^{i+1} = \Theta^i - \alpha \nabla_\Theta l$
    $i \leftarrow i + 1$
  end
end

Algorithm 1: Stochastic Gradient Descent (SGD).

splitting it into 100 batches of 10 samples each would be considered batch gradient descent. Once done, all the free parameters are updated and the process repeats for the next epoch over the whole training set.

When using the loss function in Equation 3.9 and the batch gradient descent algorithm, the optimization process only requires the calculation for the gradients $\nabla_\Theta l$. The gradients are obtained using the back-propagation algorithm, which is the result of applying the chain-rule starting from the output y all the way back to the input $\mathbf{x}$. First, the forward pass is calculated by forward-propagation of the input signal $\mathbf{x}$, as described in Equations 3.4-3.6. The output gradient is shown as:

$$\frac{\partial}{\partial f(x)_c} \left[-log \left[f(x)_y\right]\right] = \frac{-\delta(y, c)}{f(x)_y} \tag{3.11}$$

A common choice for the output in the context of classification is the Softmax layer. This layer is defined as an output neuron for which the activation function is the softmax function in Equation 3.12, where $\mathbf{a}$ is the pre-activation, y is the output class and c' is an index over all the possible output classes. For example, for model with two possible outputs $y \in \{0, 1\}$.

$$softmax(\mathbf{a})_y \equiv \frac{e^{\mathbf{a}_y}}{\sum_{c'} e^{\mathbf{a}_{c'}}} \tag{3.12}$$

The softmax layer pre-activation partial derivative is shown as:

$$\frac{\partial}{\partial a^{(H+1)}(x)_c} \left[-log \left[f(x)_y\right]\right] = -(\delta(y, c) - f(x)_c) \tag{3.13}$$

The gradient calculation of the hidden layers and their pre-activations are is shown in equations 3.14-3.15, when $\odot$ denotes element-wise product. The partial derivatives $g'(a)$ for common activation functions $g(a)$ are shown in Table 3.1.

$$\nabla_{h^{(k)}(x)}\left[-log\left[f(x)_y\right]\right] = W^{(k+1)^T}(\nabla_{a^{(k+1)}(x)}\left[-log\left[f(x)_y\right]\right]) \tag{3.14}$$

$$\nabla_{a^{(k)}(x)}\left[-log\left[f(x)_y\right]\right] = (\nabla_{h^{(k)}(x)}\left[-log\left[f(x)_y\right]\right])\odot\big(g'(a^{(k)}(x)_1), ..., g'(a^{(k)}(x)_j), ..., g'(a^{(k)}(x)_{N_k})\big) \tag{3.15}$$

In order to accelerate learning, a physically inspired momentum term [65] is added to the gradient updates. This additional hyperparameter μ could be considered as the mass of the optimization process who tends to continue going towards the same direction due to its momentum. When using momentum, the update rule becomes a two-step process; first we update the velocity v_t using the learning rate ϵ, as in Equation 3.16, then we update the parameters Θ_t in Equation 3.17 [84].

$$v_{t+1} = \mu v_t - \epsilon\nabla f(\Theta)|_{\Theta_t} \tag{3.16}$$

$$\Theta_{t+1} = \Theta_t + v_{t+1} \tag{3.17}$$

Another option is to use the Nesterov Momentum [84] update, which first propagates the parameters and only then modifies the velocity:

$$v_{t+1}^{nesterov} = \mu v_t - \epsilon\nabla f(\Theta)|_{\Theta_t+\mu v_t} \tag{3.18}$$

The difference between classical momentum and Nesterov momentum is shown in Figure 3.6.

In order to train large neural networks with high model capacities (which is common terminology for describing that it can model many different functions accurately) and still obtain low-variance optimization solution that does not overfit the training data, we use a regularization technique called early stopping. The training data is split into three sets; (a) training, (b) validation, and (c) testing set. The training set is used directly for the optimization process, while measuring the negative log-likelihood and error-rate over both the training and the validation sets. While the optimization algorithm is running, we always keep a copy of the parameters that achieved the highest accuracy over the validation set, hoping that it reliably represents the expected results over yet unseen data from the testing set. In this regularization technique we take the set of model parameters that obtained the best performance over the validation set during the optimization process, and it is named

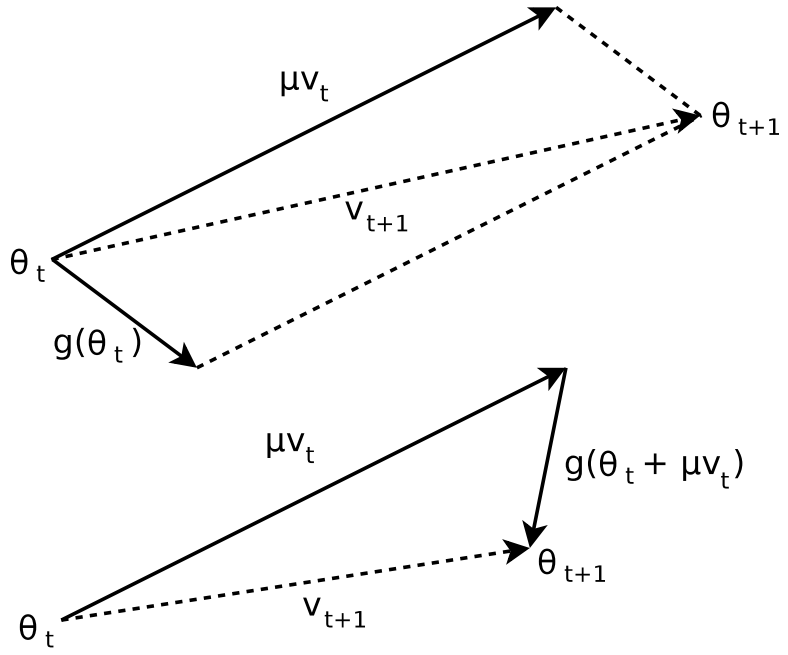


Figure 3.6 Classical versus Nesterov Momuntum, courtesy of [84]. On the top, we see the classical momentum update, calculating the gradient at the current location in parameter-space. At the bottom, Nesterov momentum is shown a measurement of the gradient g only after taking a step of μv_t in parameter-space.

early stopping because this is equivalent to stopping the optimization process when the best performance over the validation set was achieved. Early stopping is probably the most commonly regularization method in deep learning, and is popular due to its simplicity and effectiveness [13]. It is simple to implement when tracking the model performance during the training process, and it is effective due to nature of cross-validation on which it relies. Early stopping could be proven to be equivalent to L2 regularization over the weights in the case of a simple linear model with a quadratic error function when using gradient descent [13] as shown in figure 3.7.

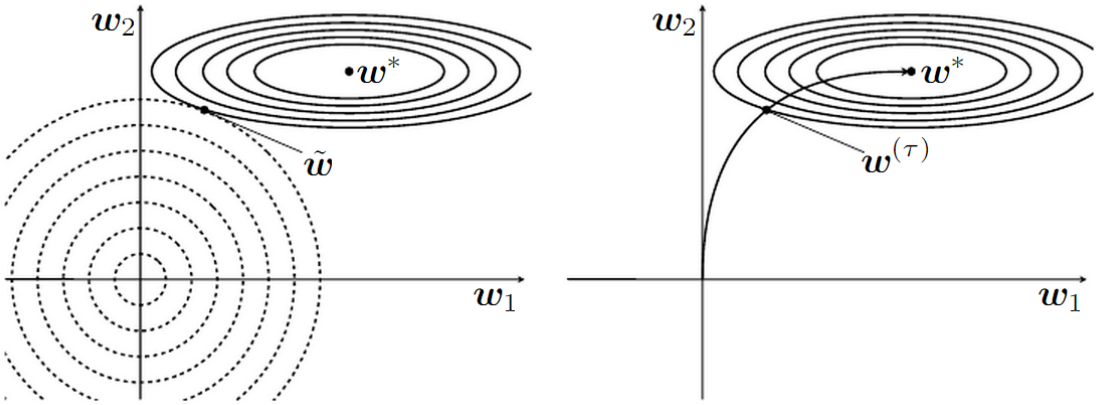


Figure 3.7 L2 Regularization versus Early Stopping, courtesy of [13]. The ellipses represent the cost function that is being optimized on the right, and both the cost function and the regularization term on the left hand side. w_1 and w_2 are the parameters that are optimized with respect to the objective function. $w*$ is the optimal configuration without regularization, and w^τ and $\tilde{w}$ are the solutions when optimising using early stopping and L2 regularization, respectively. On the left hand side we see the behaviour of L2 regularization over the weights w_1 and w_2, on the right hand side we see the optimization process as the curve starting at the origin using Early Stopping achieving an identical result.

This chapter discussed the mathematical background of convolutional neural networks, and this framework will be used in the next chapters to design a model for Multiple Sclerosis lesion segmentation. The building blocks of the proposed method of this thesis are mainly convolutional neural networks. They are designed using different configurations of layers and activation functions, and trained using the algorithms, while exploring different settings for their hyperparameters that were discussed along this chapter. The next chapters will present the details of the proposed technique and CNNs, and will be followed by the different

experimental configurations and their results.

Chapter 4

Deep Learning for MS Lesion Segmentation

The method presented in this thesis is designed to detect and delineate Multiple Sclerosis lesions in brain MR images. These input images are multi-modal MRI, and therefore composed of several different modalities each (e.g. FLAIR, T1p, T2w, etc.). This task is mainly a detection task, in which the lesions are detected, but also a delineation task for defining their boundaries. It is challenging because a well-established set of manually designed features that capture MS lesions does not currently exist. For this reason, the proposed method is designed to automatically and objectively extract such features by using deep learning techniques. In order to embed context information within the framework, the method uses a multi-scale, hierarchical segmentation approach that employs convolutional neural networks (CNN) and multi-layer perceptron (MLP) models as feature extractors, voxel-level classifiers and full lesion detectors.

First, the input MRI modalities are fed into a set of parallel local voxel classifiers, analyzing different neighbourhood resolutions. The purpose of this layer is to extract features and characterize lesion in terms of local intensities, and embed context using local and larger neighbourhood information. The results of the first level are then augmented using prior spatial information regarding healthy tissues and pathology, and followed by a second level of classification. The purpose of the second level is to make a more informed decision, which is based on the results of the three Level-1 classifiers jointly. This multi-scale approach was designed in order to use the decisions from the first level and further model them as an ensemble. For example, if the local voxel intensity appears to correspond to a lesion, but the larger-neighbourhood context seem to be of a healthy tissue, this classifier is designed to

model the correct output classification. The inputs to the second level are the results from the first level and the spatial priors that were registered to the T1-weighted image of each subject, and provide additional contextual information for the decision. Three different architectures were designed to address the task from this stage until the final output, including an artificial neural network and two random forests.

The first architecture models the joint distribution using an ANN. The second architecture further processes the results using a third level of lesion-wise classification. The purpose of this additional level is to eliminate false positive detections of lesions from the previous stage by reconsidering the fully segmented lesions on a per-lesion basis. The classifier is designed to learn how to detect false positives of the previous level. The first and second architectures are shown in Figure 4.1.

The third and fourth architectures model the joint distribution using RFs. The reason to design and test such classifiers is their success in the field of MS lesion segmentation [32]. This success was obtained, however, using hand-crafted features, and it would be interesting to explore if this concept works when using features that were automatically learned by the CNNs of the first level. These architectures are shown in Figures 4.2 and 4.4.

The neighbourhoods used for all classifiers presented in this thesis are three-dimensional, and therefore include information from adjacent brain slices in order to reach more informed decisions between lesion and non-lesion voxels. The different neighbourhoods used for Level-1 classifiers are illustrated in Figure 4.3.

4.1 Level-1

The purpose of the first level of classifiers is to learn and model several lesion characteristics:

- Voxel intensity
- Close-neighbourhood information
- Larger-neighbourhood model

The first classifier, CNN1-1, is designed to model the intensity of lesion voxels using a neural network classifier. In order to maintain a higher degree of robustness to local noise and acquisition-related artifacts, the classifier learns to discriminate lesions while augmenting the information provided by voxel intensities with information from its immediate neighbours over all input modalities. For example, a single hyper-intense voxel within a strictly hypo-intense

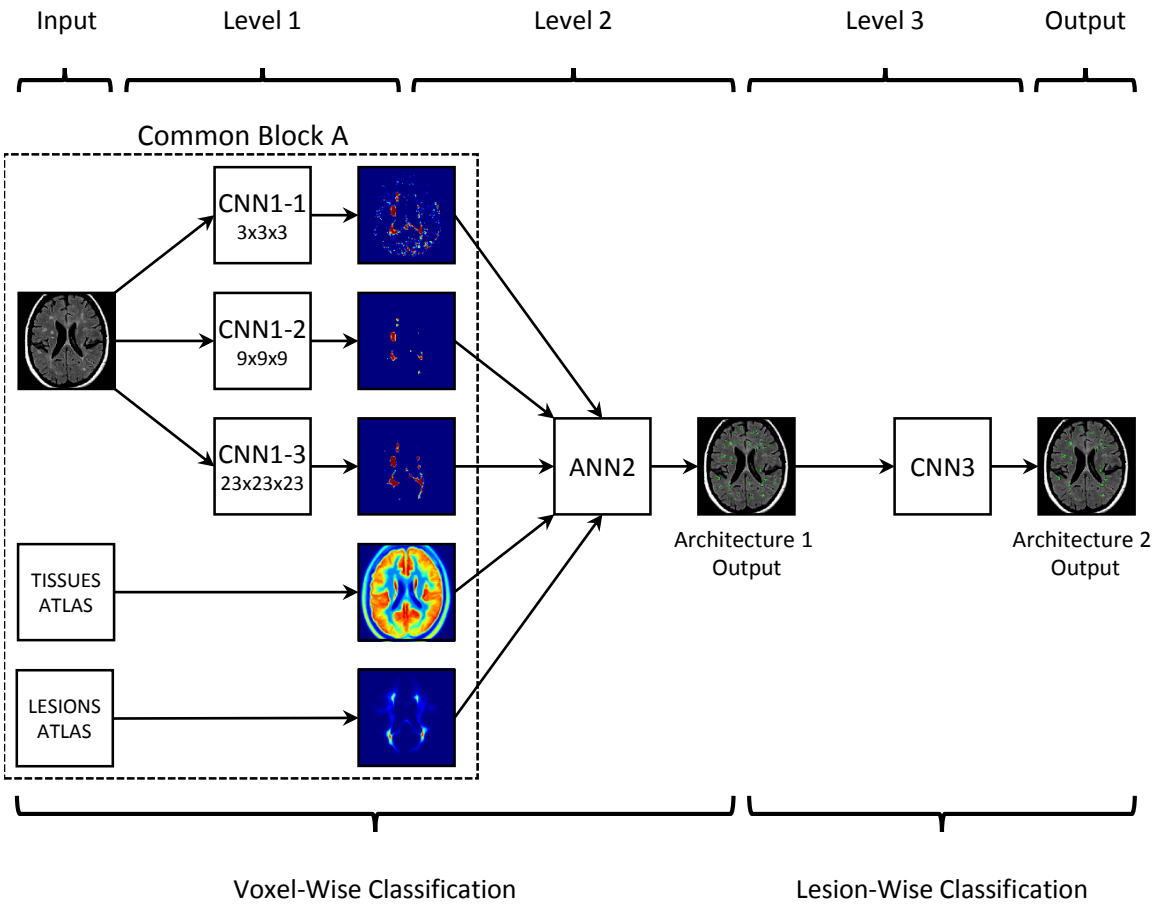


Figure 4.1 Architectures 1 and 2 Illustration. The method assumes to get a series of multi-contrast images (e.g. FLAIR, T1-, T2-weighted, PD and T1 post-contrast) as input modalities, augmented with healthy-tissue and lesion atlas priors. At level 1, the MR images are classified using 3 scales of 3D convolutional neural networks of neighbourhood sizes 3x3x3, 9x9x9 and 23x23x23. The results are then concatenated to the atlas and lesion spatial priors in order to form the input for level 2, in which a trained ANN classifies the multi-scale augmented feature vectors. At level 3, the voxels are segmented to create 3D lesions, then each candidate lesion is classified to form the final output. Notice that Common Block A is also used in Architectures 3 and 4.

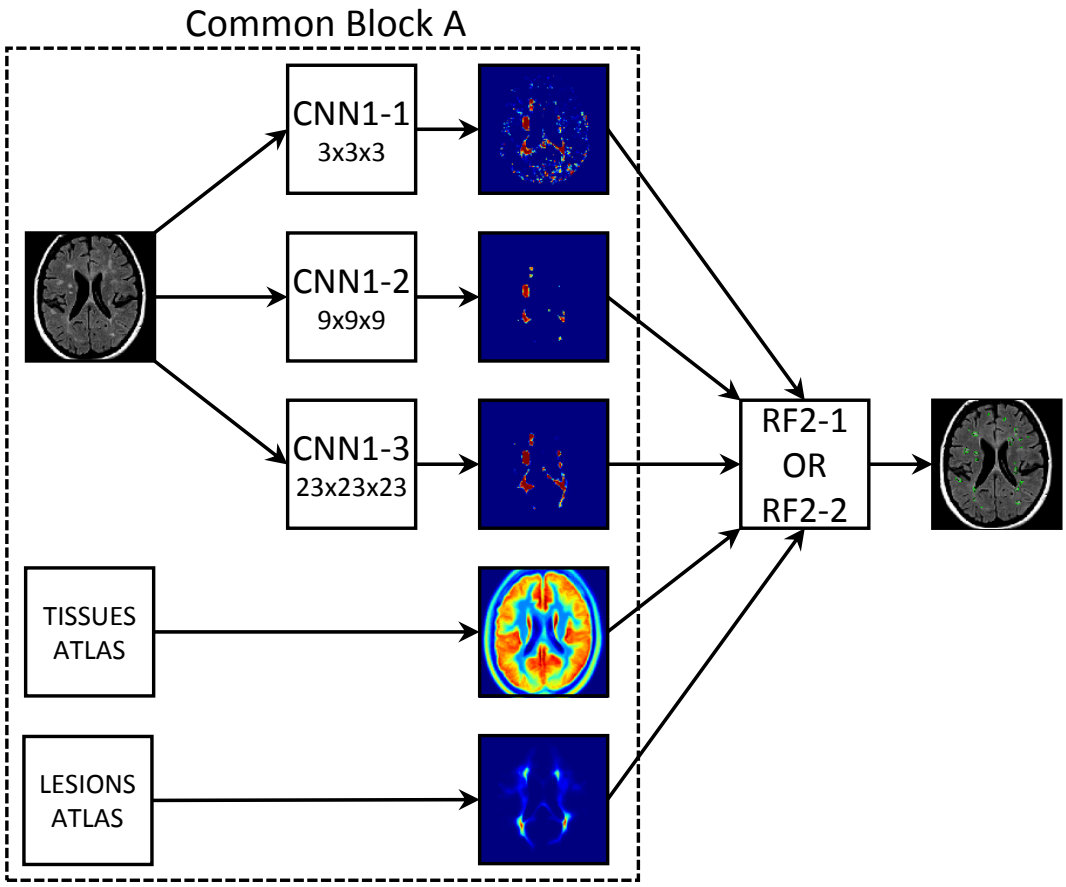


Figure 4.2 Architectures 3 and 4 Illustration. The method assumes to get a series of multi-contrast images (e.g. FLAIR, T1-, T2-weighted, PD and T1 post-contrast) as input modalities, augmented with healthy-tissue and lesion atlas priors. At level 1, the MR images are classified using 3 scales of 3D convolutional neural networks of neighbourhood sizes 3x3x3, 9x9x9 and 23x23x23. The results are then concatenated to the atlas and lesion spatial priors in order to form the input for level 2, in which a trained RF classifies the multi-scale augmented feature vectors. Notice that Common Block A is also used in Architectures 1 and 2.

neighbourhood might indicate either a lesion, or an artifact. This model has the capacity that enables it to automatically learn to differentiate these two cases from the training data, and model the local noise in order to achieve rather informed resolutions.

In the framework, the design of the voxel-intensity classifier is not intended to model the complete neighbourhood of the voxel. It is designated to consider only the underlying intensity, and extract features that describe it without modeling the surroundings of the voxel. Considering the two additional first-level classifiers, this will prevent an undesirable coadaptation (learning the same features in different classifiers) of the extracted features, and enable a second-level classifier to learn more robust joint distributions over the multi-scale ensemble.

The second first-level classifier, CNN1-2 represents a medium-range neighbourhood model. At this range, delineation of lesions is expected to become a slightly harder task, and the determination of the exact lesion boundaries would rely more on the context of the close neighbourhood than its underlying intensity model from CNN1-1. This classifier is introduced to a larger input size, but not too large, in order to prevent coadaptation with the third first-layer classifier.

The last first-layer convolutional neural network, CNN1-3 is designed to model a large/long-range local neighbourhood around the voxel. The purpose is to extract contextual information into features that will help discriminate the lesions based primarily on the context rather than its own appearance. Modeling large neighbourhoods of voxels could aide detecting lesions based on their shapes, high-level appearance, location, anatomical structure, inter-lesion statistics and more. Learning these relationships from the training data is expected to have a positive effect on the quality of the results in terms of false-positive detections of structures that exhibit lesion-like characteristics from close, but are located in either anatomically-impossible locations for lesions or form a shape that do not comply with the model for the shapes formed by MS lesions.

4.2 Level-2

The next level of the method, Level-2, is using the predictions from Level-1 and augments them with registered spatial priors for tissues such as white-matter, gray-matter, cerebrospinal-fluid, and lesions. In order to combine Level-1 predictions and obtain the final segmentation, two different approaches were proposed:

- Artificial Neural Network (ANN), used in architectures 1 and 2.

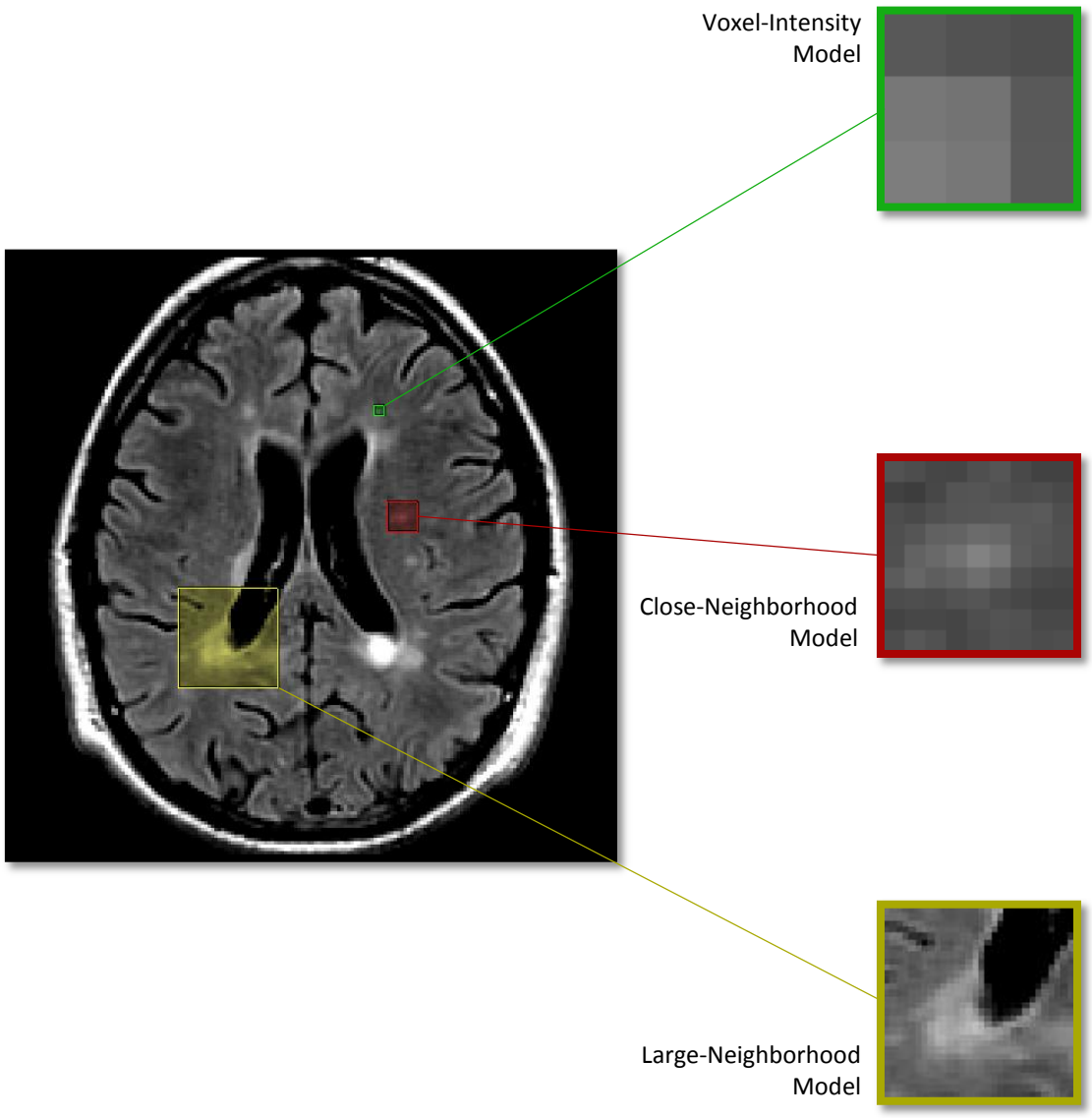


Figure 4.3 Level-1 Neighbourhood Models. This figure illustrates the multi-scale approach used in Level-1 of the method. Three different sizes neighbourhood models are trained from the data in order to learn discriminative features of multiple sclerosis lesions, based on: (a) voxel-intensity, (b) close-neighbourhood, and (c) large-neighbourhood. We can see the context information in the large-neighbourhood patches (yellow), the local information in the close-neighbourhood aperture in red, and local intensities in green. All the models in this thesis use 3D neighbourhoods, and are shown in 2D for illustrational purposes only.

- Random Forest (RF), used in architectures 3 and 4.

The first and second architecture use an ANN to combine the prediction-information and spatial priors (see Figure 4.1). The purpose is to build a joint distribution over them and learn it from the training set. In order to compare the performance to an additional type of classifier, the third and fourth proposed architectures are using Random Forests (RFs) to model the same joint distribution instead of an artificial neural network (see Figure 4.2). Since the data will contain a small amount of features at this stage, it is expected to obtain comparable results to the ANN, principally due to the fact that this classifier operates on a bag of voxels, for which the neighbourhood-context was already extracted and presented as features by the three CNNs in Level-1.

4.3 Level-3

Architecture 2 defines an additional convolutional neural network, which is named CNN3 in Figure 4.1, and is designed to reduce the false-positive rate directly by eliminating lesion candidates from Level-2. This structure ensures that the decisions will be made based on a the characteristics of the Level-2 classifier, and therefore will be customized to the types of false- and true-positives that are typically introduced by the trained Level-2 ANN. This last level could be overseen as a postprocessing stage that involve decisions that are made at the lesion-level rather than voxel-level.

Given the fact that a weighted average over the last hidden layer activations of CNN1-1, CNN1-2 and CNN1-3 forms the final classification, it could be interesting to visualize these features. As shown in Equation 3.6, and considering the usage of a Softmax unit for output layer activations as described in Equation 3.12 above, the inputs are a multiplication of the weights matrix and the last hidden layer activations, which in the binary case reduces to vector element-wise multiplication. Therefore, one can examine the weights based on their norm value, in which large positive values should indicate highly discriminative features towards the lesion class, while negative values could pinpoint feature-maps that negate possible presence of lesions.

This chapter described the four proposed architectures for automatic detection of MS lesions using CNNs, ANNs and RFs. In the next chapter, we will present the experimental setup and different configurations for each of the processing levels. Also, the performance of the different architectures will be compared, and the features learned by the first level of convolutional neural networks will be shown. The chapter will start by describing the data

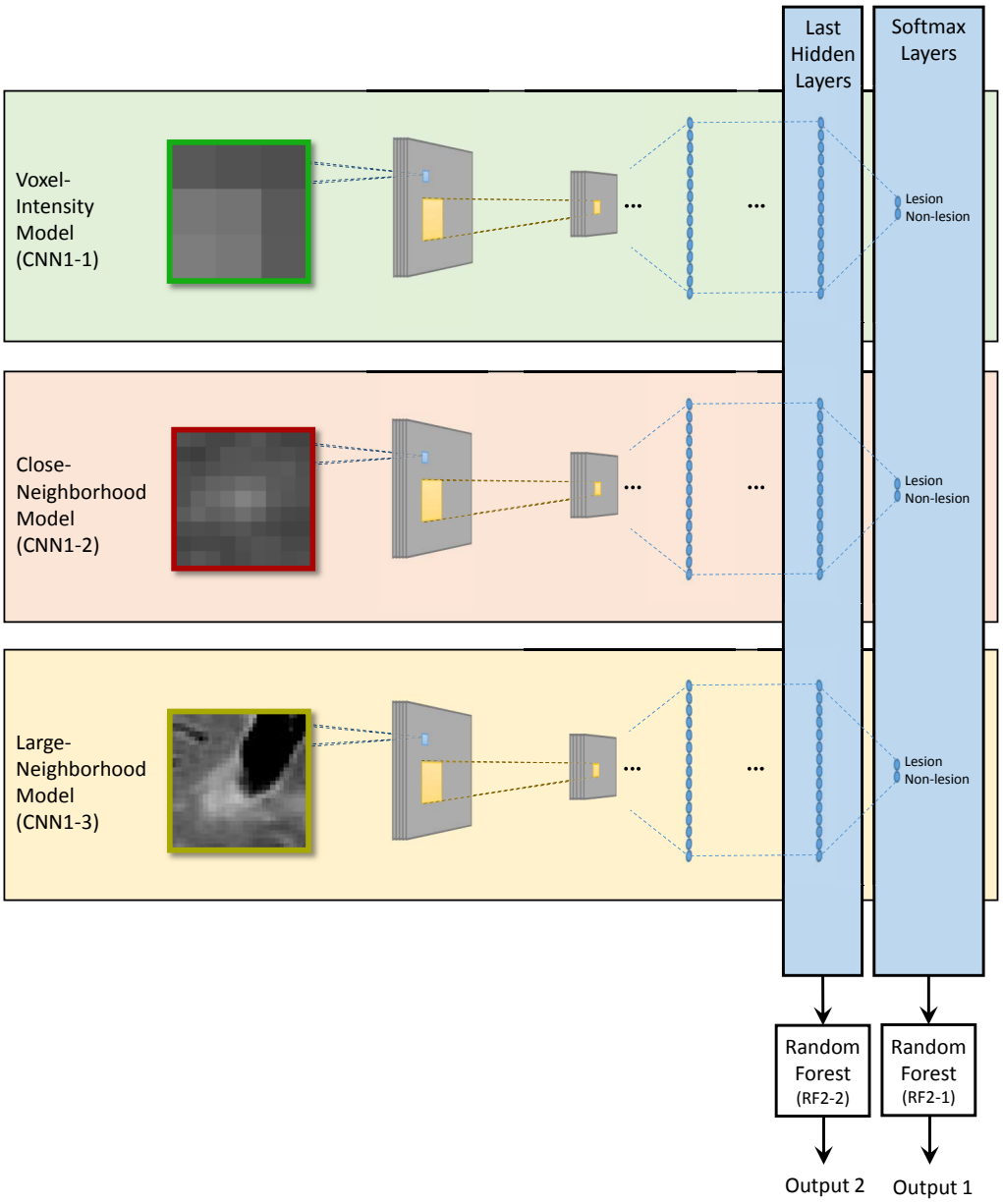


Figure 4.4 Random-Forest Joint Distribution Model for Architectures 3 and 4. The three Level-1 CNN classifiers output their predictions at their last Softmax layer. These predictions are used as input features for a random forest RF2-1 that models a multi-scale joint distribution. Their last hidden layer activations, which were only used internally, are exposed and used as features for RF2-2. All the models in this thesis use 3D neighbourhoods, and are shown in 2D for illustrational purposes only.

used for the experiments, preprocessing procedure and balancing technique, and continue with the evaluation metrics and the results.

Chapter 5

Experiments

This chapter presents the experimental setup and the results of the proposed architectures over the data, addressing the challenge of Multiple Sclerosis lesion segmentation using deep learning techniques. In this chapter, the data used for the experiments, taken from large clinical trials, will be described in detail, followed by its pre-processing. The challenge of sampling data for training the proposed architectures using these large clinical datasets will be presented, followed by the evaluation metrics in the contexts of detection and segmentation. Once the ground is set, the results from each of the stages of the architecture will be presented over several different configurations for their internal parameters over Dataset A. These configuration will then be tested over an additional and much larger clinical trial, Dataset B. Also, the features extracted automatically using the convolutional neural networks will be explored and presented to the reader both as vectors and images that will be discussed in details.

5.1 Data

The data used to train and test the proposed method is based on two proprietary multinational and multicenter real clinical trials. The dataset that was used for most of the reported experiments, Dataset A, contains 112 different Multiple Sclerosis patients with T1-, T2-weighted, T1-post-contrast, proton-density (PD) and fluid-attenuated inversion recovery (FLAIR) modalities for each. The trial was randomized, and conducted over 37 sites. The "ground truth" manual labeling was performed in a semi-automatic manner by trained experts. First, the images were automatically classified using a Bayesian method [27] that considers three out of the five available modalities. The resulting labeling is passed through

a post-processing stage, in which certain pre-defined heuristics are automatically applied in order to remove certain false-positive detections. These results are then delivered to MRI readers that are trained to follow a strict industrial labeling protocol in which they manually validate the output, while maintaining minimal human variability. The human raters are only allowed to completely remove a lesion object in the case of false detections, and are not allowed to modify the delineation of the objects or to add undetected pathology.

The proposed architecture and its hyper-parameters were optimized over Dataset A only, and a second dataset was used to test if it generalized to a different, and much larger, clinical trial dataset. The second proprietary clinical trial, Dataset B, was conducted over a 24-months period on a population of 1063 Multiple Sclerosis patients from 171 sites. The lesion labeling followed the same protocol described above for Dataset A, and provided the same five modalities: T1-, T2-weighted, T1-post-contrast, PD and FLAIR.

5.2 Pre-processing

The data used for the experiments was registered to a common-space, corrected for non-uniformity bias field, intensity-normalized [62] and skull-stripped. First, the images are registered into the stereotaxic space, followed by an ICBM152 healthy-brain atlas registration. The atlas was built using 152 manually labeled healthy subjects and registered using ANIMAL [19][21], and provides the pipeline with spatial priors for: (a) white matter, (b) gray-matter, and (c) cerebrospinal fluid. Lesion priors were generated by non-linearly registering automatically-segmented, expert-corrected annotations over a different clinical trial that contains 3714 relapsing-remitting MS subjects [25]. Additionally, the images were processed for intensity normalization and bias correction using the N3 algorithm [76]. The skull-stripping was done using the Brain Extraction Tool (BET) [77].

5.3 Sampling to Balance the Classes

Voxels labeled as Multiple Sclerosis lesions constitute only a small fraction of the overall data. In order to train a convolutional neural network that successfully classifies them without becoming overwhelmed by the non-lesion training data, the dataset must be balanced. The technique for balancing the dataset was designed while taking the following considerations into account:

Overall Balance The dataset should be kept balanced as a whole.

Location-Wise Balancing High degree of balancing should be maintained for every location within the brain image. For example, a specific location for which 90% of the patients are positively labeled as a lesion should be provided with sufficient amount of counter-examples that will prevent the CNN from always classifying this location as MS lesion.

Preserving Positively-Labeled Samples Since positive samples are rare, we should avoid dropping them as much as possible during the balancing procedure.

Strictly-Negative Locations The technique should sample from locations in which MS lesions do not exist at all in the dataset. This will train the network to classify them correctly, even at the cost of violating the location-wise balance described above.

To provide the classifiers with balanced representations from each label, the brain images were first divided into three different sections, according to the empirical lesion appearance over all regions: (a) positive-only, (b) mixed, and (c) negative-only, where positive/negative refer to the type of labels that occur within each region. The empirical lesion frequency over the training set is shown in Figure 5.1. Once the brain was divided into regions, the sampling algorithm generates a spatial sampling map that is based on balancing the mixed regions, while also sampling from the negative- and positive-only regions. The balancing of the mixed regions is aimed to prevent classifying regions with high lesion empirical frequency, for example, around the ventricles, from being learned as lesions and resulting in false-positive classifications. On the other hand, regions with low frequency would to suffer from the inverse phenomena, being classified as non-lesion, which is prevented using this sampling mechanism. Another important characteristic of the sampling framework is that it also balances between negative-only regions, which contain only non-lesion examples in the training set. Sampling neighbourhoods from this region will help the classifier learn them as negative samples instead of unknowns, and therefore the learned lesions-manifold will take them into consideration as it forms during the training process.

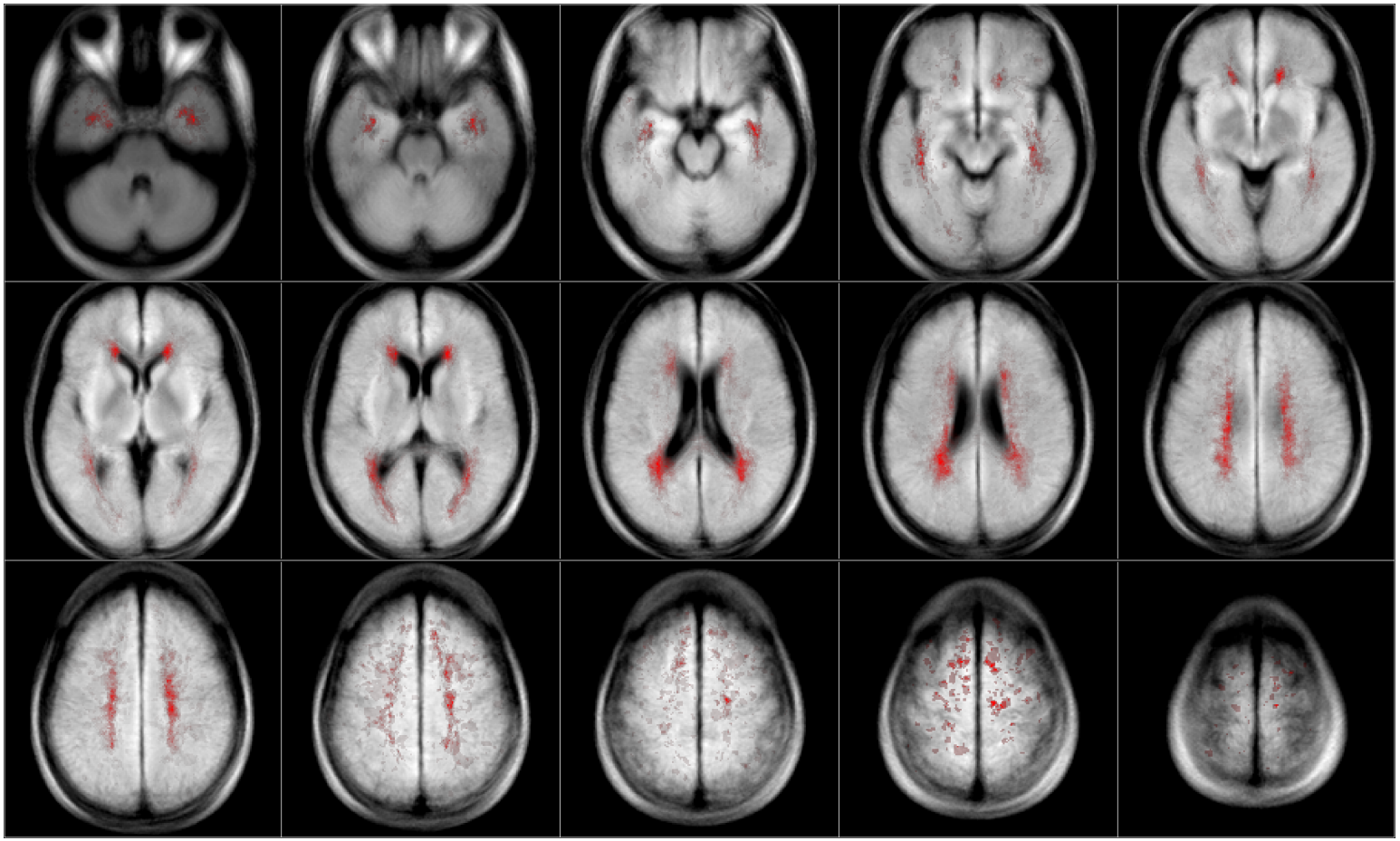


Figure 5.1 Empirical Lesion Frequency. High spatial frequency of lesion samples over Dataset A is shown in red. In order to balance the dataset, this spatial imbalance between positive and negative examples is handled within the sampling framework. The slices shown left-to-right, top-to-bottom are 14, 16, 18, 20, 22, 24, 26, 29, 31, 33, 35, 37, 39, 41 and 44.

5.4 Evaluation Metrics

The evaluation of lesion segmentation methods is based on metrics that quantify its similarity to the ground truth labels. There exist several different methods to measure such similarities, and in general they are divided into two major classes:

- Voxel-Wise Metrics
- Lesion-Wise Metrics

Voxel-wise metrics measure the correlation between classified voxels in the segmentation results and the ground truth labels on a per-voxel basis. This way, the label of every voxel in the image is compared between the two, graded according to a pair-wise similarity metric, and finally aggregated to obtain an overall labeling similarity score. In order to understand these scores, it is essential to examine all the possible scenarios between two labeled images A and B, as shown in Figure 5.2. In the case that a voxel is predicted as positive while the truth labeling is negative we consider it as False-Positive (FP). The inverse, when a voxel is predicted as negative while the truth labeling is positive is considered a False-Negative (FN). True-Negative (TN) is the case in which both labeling are negative, and True-Positive (TP) is when both are positive. We are now ready to define true- and false-positive rates (TPR/FPR), as shown in Equations 5.1-5.2. True-Positive Rate is also referred to as sensitivity. The Dice score for similarity between sets is shown in Equation 5.3, and is a normalized measurement for the positive-label overlaps between two sets. The Positive Predictive Value (PPV) is the rate between true positives to total positives, and is defined in Equation 5.4.

$$TPR = \frac{TP}{TP + FN} \tag{5.1}$$

$$FPR = \frac{FP}{FP + TN} \tag{5.2}$$

$$DICE = \frac{2TP}{(TP + FP) + (TP + FN)} = \frac{2TP}{2TP + FP + FN} \tag{5.3}$$

$$PPV = \frac{TP}{TP + FP} \tag{5.4}$$

Voxel-wise metrics are widely used in the literature [87][7][9] for measuring the performance of healthy tissue (and some pathology) segmentation methods. Nonetheless, our primary goal is detecting all the lesions, even the small ones, and in this context voxel-wise

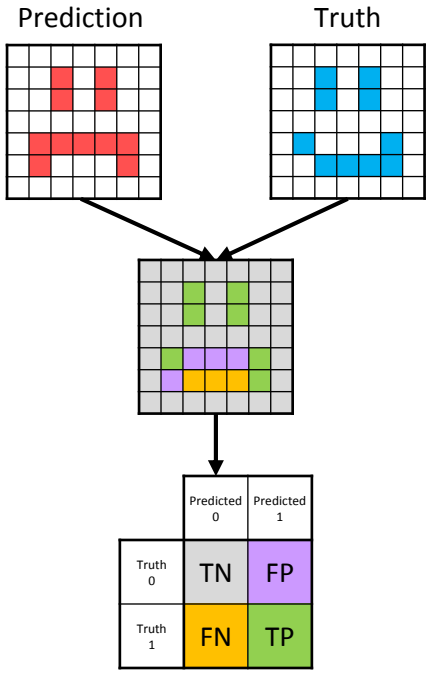


Figure 5.2 Voxel-Wise Confusion Matrix. Two binary labeled images A and B, who correspond to the prediction and the ground-truth labeling are voxel-wise tested for correspondences between the voxels. True-Negative (TN) is the case in which both labeling are negative, True-Positive (TP) is when both are positive. If the predicted label is positive and the truth label is negative, it is considered False-Positive (FP). If the predicted label is negative but the truth label is positive, it is called a False-Negative (FN).

metrics do not correctly penalize errors due to missed small lesions. When segmenting lesions, the output is the delineation of binary objects (BLOBs). A BLOB-, or lesion-wise score metric would ideally penalize on a per-lesion basis, and has the advantage of considering all lesions equally for the similarity grading process. For example, missing a very small lesion would hardly be penalized on a per-voxel metric, but would be equally penalized to a large lesion when using such a lesion-wise metric. In order to obtain an additional set of detection metrics, to be used besides the voxel-wise metrics, we generalize the previously shown metrics from Equations 5.1-5.4 to lesion-wise segmentation, and use the following definitions:

Lesion BLOB Definition Lesions BLOBs are defined as a set of adjacent neighbouring voxels, while adjacency is defined using the 18-connectivity neighbourhood as shown in Figure 5.3.

Lesion Overlap Criteria A lesion BLOB from segmentation A overlaps with segmentation B if and only if: (a) three or more of the BLOB voxels are labeled positive in segmentation B, or (b) at least 50% of the BLOB voxels are labeled positive in segmentation B. This criteria is based on clinical trial protocol.

Lesion-wise True-Positives (LTP) Count of the ground-truth lesion BLOBs that overlap with the lesions defined by the segmentation results.

Lesion-wise False-Positives (LFP) Count of the segmentation results lesion BLOBs that do not overlap with the ground-truth labels. This term is also referred to as False Discovery Rate (FDR).

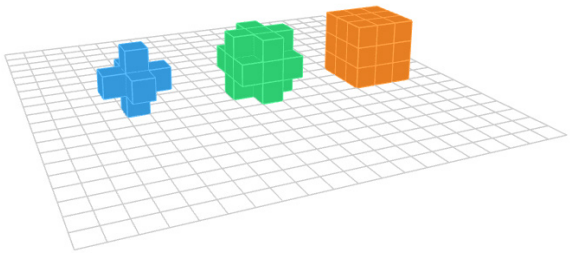


Figure 5.3 Voxel Connectivity Models. From left to right: (a) 6-neighbours-, (b) 18-neighbours-, and (c) 26-neighbours-connectivity models with respect to the central voxel.

Using the definitions above, the Lesion-wise True Positive Rate (LTPR), Lesion-wise False Positive Rate (LFPR) and Lesion-wise Positive Predictive Value (LPPV) are defined in Equations 5.5-5.7.

$$\text{LTPR} = \frac{\text{LTP}}{\text{Total ground truth lesion count}} \tag{5.5}$$

$$\text{LFPR} = \frac{\text{LFP}}{\text{LFP+LTP}} \tag{5.6}$$

$$\text{LPPV} = \frac{\text{LTP}}{\text{LFP+LTP}} \tag{5.7}$$

Both voxel-wise and lesion-wise metrics are important for the task. The lesion-wise metrics will quantify the performance of the detection task, while the voxel-wise metrics will measure the quality of the delineation of lesion.

The last metric used for evaluation of the classifiers is their misclassification rate. Misclassification rate is the rate of incorrect classifications, as defined in Equation 5.8.

$$\text{Misclassification Rate} = \frac{\text{TP+FP}}{\text{TP+FP+TN+FN}} \tag{5.8}$$

5.5 Level-1 Results

Dataset A was divided to a training set of 60 images, a validation set of 24 images, and a testing set of 28 images which were used only for the final testing. The input MRI modalities are sampled for neighbourhood patches of sizes defined by the 3D CNN classifiers of the first layer. Each patch is then learned as a positive neighbourhood example of the central voxel if and only if it is labeled as lesion in the training target images. An ensemble of CNN classifiers of the following sizes were trained for the task:

- 3x3x3 Voxels
- 9x9x9 Voxels
- 23x23x23 Voxels

Examples of the middle axial slice from the training dataset for neighbourhoods of dimensionality 23x23x23 are shown in Figure 5.4.

The 3 different classifiers were cross-validated for several different CNN architectures and optimization hyper-parameters, while the architecture used for the 9x9x9 neighbourhood yielded the best misclassification results of 4.94% over the balanced validation data, using the following layers: (a) convolution using 128 kernels of size 4x4x4, (b) max-pooling over 2x2x2

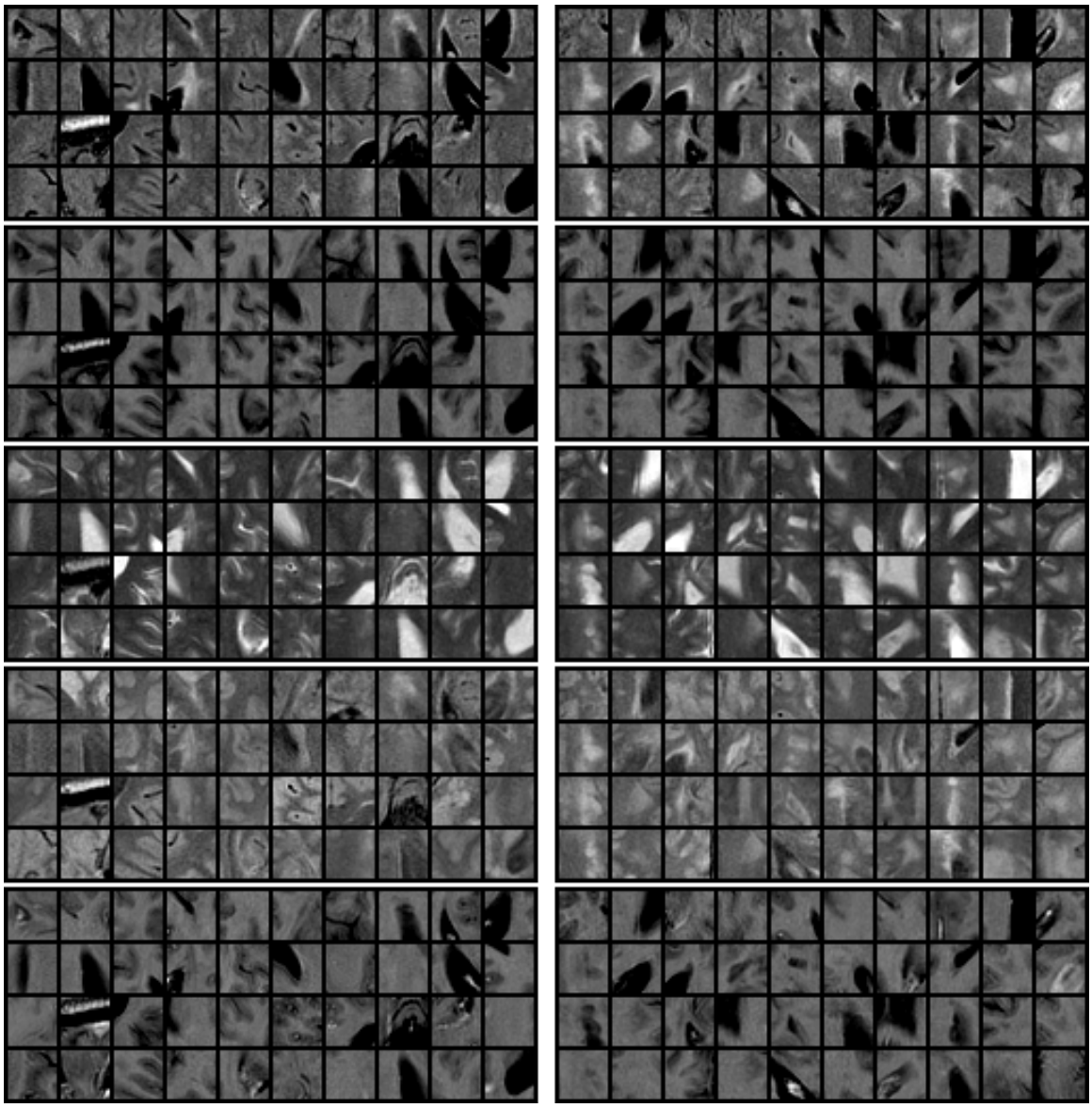


Figure 5.4 Layer-1 Neighbourhood Examples. The images are the middle axial slice of the MRI examples used to train the first level of convolutional neural network classifiers over Dataset A. The patches shown are of neighbourhood size 23x23x23 voxels, negative examples (non-lesions) on the left-hand side and positive examples (lesions) on the right-hand side. The image shown depicts 40 different examples for both positive and negative examples, grouped from top to bottom: FLAIR, T1-weighted, T2-weighted, PD and T1-post-contrast examples.

regions, (c) drop-out, keeping 70% of the original output and multiplicative augmentation factor of 1.5 (d) convolution using 128 kernels of size 2x2x2, (e) max-pooling over 2x2x2 regions, (f) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, (g) ReLU layer with 2048 neurons, (h) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, (i) ReLU layer with 2048 neurons, (j) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, and (k) softmax layer with 2 output neurons. The optimization technique used was batch gradient descent with learning rate of 0.001 and momentum of 0.1 as hyper-parameters.

The best found architecture for the 23x23x23 neighbourhoods, yielding 5.76% misclassification rate over the balanced training set, was as follows: (a) convolution using 128 kernels of size 3x3x3, (b) max-pooling over 3x3x3 regions, (c) convolution using 128 kernels of size 2x2x2, (d) max-pooling over 2x2x2 regions, (e) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, (f) ReLU layer with 1024 neurons, (g) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, (h) ReLU layer with 1024 neurons, (i) drop-out, keeping 50% of the original output and multiplicative augmentation factor of 2.0, and (j) softmax layer with 2 output neurons.

As for the 3x3x3 neighbourhoods, the best found architecture, achieving 6.69% misclassification rate was: (a) ReLU layer with 1024 neurons, (b) ReLU layer with 1024 neurons, and (c) ReLU layer with 1024 neurons.

The number of examples used for training/validating the 3x3x3, 9x9x9 and 23x23x23 architectures were 1,352,024/95,222, 1,348,780/95,222 and 159,876/11,686, respectively. The results for 17 selected experimental architectures are shown in Table 5.1. As we see, the 9x9x9 neighbourhood-based CNN classifiers yielded the best misclassification rates, and these results could be attributed to two main factors. First, comparing to the 3x3x3 CNNs, the 9x9x9 could learn significantly more discriminative features as a result of observing a 27 times larger neighbourhood of 729 voxels. Second, due to memory limitations and storage drive space constraints, larger amounts of training examples were used for training the 3x3x3 and 9x9x9 architectures comparing to the amount used for training the 27x27x27.

As for the 3x3x3 patches, we can see that the misclassification rates obtained using convolutional neural networks are comparable to those obtained by using only multilevel perceptron models. This is for the obvious reason that weight sharing and convolution over 3x3x3 patches using 2x2x2 kernels is almost equivalent to a fully connected 3x3x3 kernel because it does not introduce a large amount of additional parameters to the model (see Figure 5.5), this would have not been the case for a larger input image size. In order to clearly see this similarity, it

Exp.	p23c6	p23c5	p23c4	p23c2	p23c1	p9c2	p9c1	p3c8	p3c7	p3c6	p3c5b	p3c5a	p3c5	p3c4	p3c3	p3c2	p3c1
Input	23x23x23	23x23x23	23x23x23	23x23x23	23x23x23	9x9x9	9x9x9	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3	3x3x3
L1	Conv. 3x3x3 128 N	Conv. 3x3x3 128 N	Conv. 3x3x3 128 N	Conv. 3x3x3 128 N	Conv. 2x2x2 128 N	Conv. 4x4x4 128 N	Conv. 4x4x4 128 N	Conv. 2x2x2 128 N	Conv. 2x2x2 128 N	Conv. 2x2x2 128 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	Maxout 1024 N	Maxout 48 N	ReLU 2048 N	Conv. 1x1x1 128 N
L2	MaxPool 3x3x3	MaxPool 3x3x3	MaxPool 3x3x3	MaxPool 3x3x3	MaxPool 2x2x2	MaxPool 2x2x2	MaxPool 2x2x2	MaxPool 2x2x2	MaxPool 2x2x2	MaxPool 2x2x2	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	Dropout 0.8	Dropout 0.8	ReLU 2048 N	Conv. 1x1x1 128 N
L3	Conv. 2x2x2 128 N	Conv. 2x2x2 128 N	Dropout 0.5	Conv. 2x2x2 128 N	Conv. 3x3x3 128 N	Conv. 2x2x2 128 N	Conv. 2x2x2 128 N	ReLU 1024 N	Dropout 0.8	Dropout 0.8	Dropout 0.5	ReLU 1024 N	ReLU 1024 N	Maxout 1024 N	Maxout 48 N	Softmax 2 N	Softmax 2 N
L4	MaxPool 2x2x2	MaxPool 2x2x2	Conv. 2x2x2 128 N	MaxPool 2x2x2	MaxPool 3x3x3	MaxPool 2x2x2	MaxPool 2x2x2	Dropout 0.5	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	Softmax 2 N	Softmax 2 N	Dropout 0.8	Dropout 0.8		
L5	ReLU 1024 N	Conv. 2x2x2 128 N	MaxPool 2x2x2	ReLU 1024 N	ReLU 1024 N	Dropout 0.7	Dropout 0.7	Softmax 2 N	Dropout 0.5	Dropout 0.5	Dropout 0.5			Softmax 2 N	Softmax 2 N		
L6	Dropout 0.5	MaxPool 2x2x2	Dropout 0.5	ReLU 1024 N	ReLU 1024 N	ReLU 2048 N	ReLU 2048 N		Softmax 2 N	ReLU 1024 N	Softmax 2 N						
L7	ReLU 1024 N	ReLU 1024 N	Conv. 2x2x2 128 N	Softmax 2 N	Softmax 2 N	Dropout 0.5	Dropout 0.5			Dropout 0.5							
L8	Dropout 0.5	Dropout 0.5	MaxPool 2x2x2			ReLU 2048 N	ReLU 2048 N			Softmax 2 N							
L9		ReLU 1024 N	Dropout 0.5			Dropout 0.5	Dropout 0.5										
L10		Dropout 0.5	ReLU 1024 N			Softmax 2 N	Softmax 2 N										
L11		Softmax 2 N	Dropout 0.5														
L12			ReLU 1024 N														
L13			Dropout 0.5														
L14			Softmax 2 N														
LR	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Mom.	0.9	0.9	0.9	0.9	0.1	0.9	0.1	0.1	0.1	0.9	0.9	0.9	0.1	0.5	0.5	0.1	0.1
Mis.	**5.76%**	6.66%	28.5%	7.38%	7.49%	5.64%	**4.94%**	7.10%	24.4%	19.7%	7.08%	7.16%	**6.69%**	9.60%	11.4%	8.61%	7.35%

Table 5.1 Layer-1 Training Performance Benchmark. The results reflect the misclassification rate for each of the architectures over balanced datasets that were used in the experiments for testing and validation. Exp. = ExperimentID (the naming convention is pXcY, where X is the size and Y is the experiment number) , Lx = Layer x, LR = Learning Rate, Mom. = Momentum, Mis. = Misclassification Rate.

was chosen to visualize only a single dimension rather than all the three-dimensional connections between the units. Connections that share the same color represent the weight sharing and missing connections stand for the 2x2x2 size of the receptive field. We can observe that in the one-dimensional case, we have 2 less connections, and 2 pairs of shared parameters, resulting in a total of 2 free parameters in a CNN versus 6 in a MLP model.

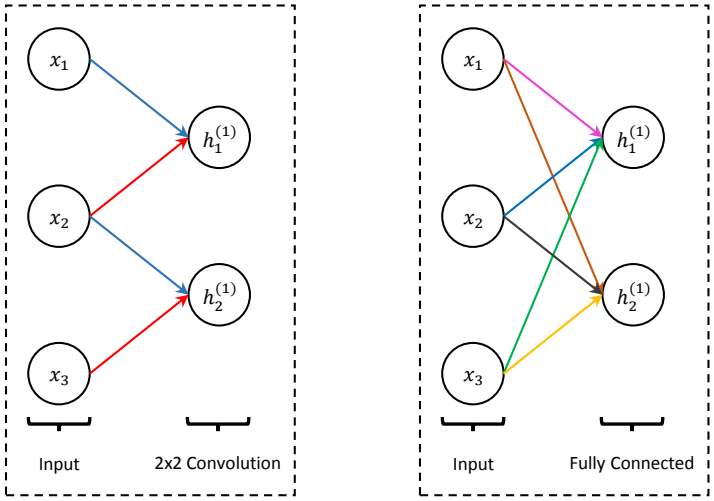
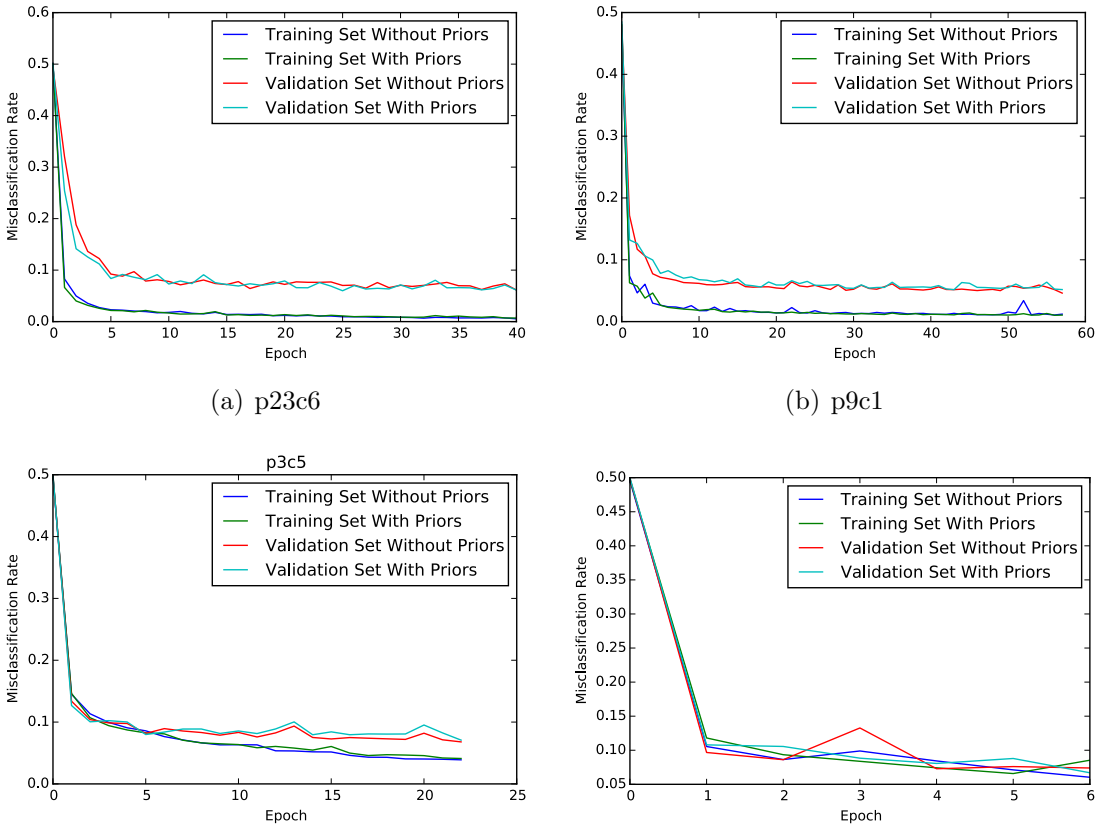


Figure 5.5 CNN vs MLP in a 3x3x3 Neighbourhood. On the left-hand side: A convolutional layer of size 2, given a size-3 input. Shared colors represent weight-sharing, and missing connections illustrate the size of the receptive field. On the right-hand side: A fully-connected layer of size 2, given an input size of 3. The figure is shown in 1D for clarity, and generalizes to the 3D model used for the experiments when x represents the input image and h its hidden representation in the neural network.

An additional experiment was performed in order to fully explore the impact of integrating the spatial ICBM152 atlas priors at the Level-1 stage of the full architecture. Prior-augmented versus non-prior-augmented models were trained simultaneously over an identical set of neighbourhood-patches (from Dataset A training and validation folds), based on several top performing architectures from Table 5.1 in terms of misclassification rate over the validation set. The training results for this experiment are shown in Figure 5.6. As we can see, the prior-augmented features misclassification rates are comparable to the rates without using the priors. As a result, it was chosen not to increase the model complexity at Level-1, and use the spatial priors only at the second stage of the classification process (Level-2) since no measurable justification was found in any of these experiments.

The qualitative and quantitative results for Level-1 are shown in figures 5.7 and 5.8-5.13 respectively for different operating thresholds, which are the thresholds over the probabilistic



(a) p23c6

(b) p9c1

(c) p3c5

(d) p3c5a

Figure 5.6 Priors Augmentation for Level-1. This figure shows a comparison of the training and validation misclassification rate along the training process, with and without spatial ICBM152 healthy tissue atlas priors. The sets of two classifiers were trained simultaneously over identical inputs in order to maintain fair correspondences along the training process. The different architectures shown are: (a) 23x23x23 neighbourhood - experimental architecture #p23c6, (b) 9x9x9 neighbourhood - experimental architecture #p9c1, (c) 3x3x3 neighbourhood - experimental architecture #p3c5, and (d) 3x3x3 neighbourhood - experimental architecture #p3c5a. The architectures are fully described in Table 5.1. The misclassification rates are for the predictions of the classifier over the training and validation sets.

output values of the CNNs from which a positive decision is taken. For example, for an operating threshold of 0.3, an output of 0.45 will be considered as a lesion and an output of 0.25 will be considered as a healthy tissue. The major issue with the classifiers at this level is their high false positive rate measured in both lesion- and voxel-wise metrics. A voxel-wise comparison between the 3x3x3 and the 9x9x9 classifiers in Figures 5.8 and 5.9 shows that the false positive rates are lower in the larger neighbourhood model. However, a careful look shows that 3 out of the 28 testing subjects were hardly affected in terms of false positive rates, and 2 others were only marginally improved. The lesion-wise comparison between the two classifiers, using Figures 5.11 and 5.12, shows a larger improvement in both the LFPR and LTPR comparing to FPR and TPR, respectively. A lesion wise comparison between the 23x23x23 and the 9x9x9 classifiers shows that when operating at very high confidence threshold (larger than 90%), the larger neighbourhood CNN achieves higher LTPR than the smaller one (0.79, 0.77, 0.68 vs. 0.71, 0.65, 0.48 for operating at 0.95, 0.97 and 0.99 respectively) at the cost of higher LFPR (0.73, 0.66, 0.51 vs. 0.80, 0.75, 0.60 for operating at 0.95, 0.97 and 0.99 respectively).

At this point, we can clearly see from the results that at Level-1 we obtained a trained ensemble of classifiers with distinctive properties. This relationship could be further modeled in a consecutive stage that utilizes their different predictions and takes advantage of their diversity. Also, it is important to notice that averaged metric results over the 28 testing subjects do not fully represent the performance of the classifier due to the very high variance between the subjects over each of the metrics as seen in Figures 5.8-5.13.

5.6 Level-2 Results

The second level of the architecture learns a joint model over the results from the previous level, while combining it with pre-registered spatial prior information regarding healthy tissues and pathology. The augmented data vector used for the first architecture consists of: (a) CNN1-1 (3x3x3) probabilistic predictions, (b) CNN1-2 (9x9x9) probabilistic predictions, (c) CNN1-3 (23x23x23) probabilistic predictions, (d) white-matter prior probabilities, (e) gray-matter prior probabilities, (f) cerebrospinal fluid prior probabilities, and (g) lesion prior probabilities. The output has two possible classes: (a) lesion, and (b) non-lesion. Data examples are shown in Figure 5.14.

The first architecture uses an artificial neural network, ANN2 in Figure 4.1, to model the joint distribution of lesions as a function of the input vectors. Recall that the the second level

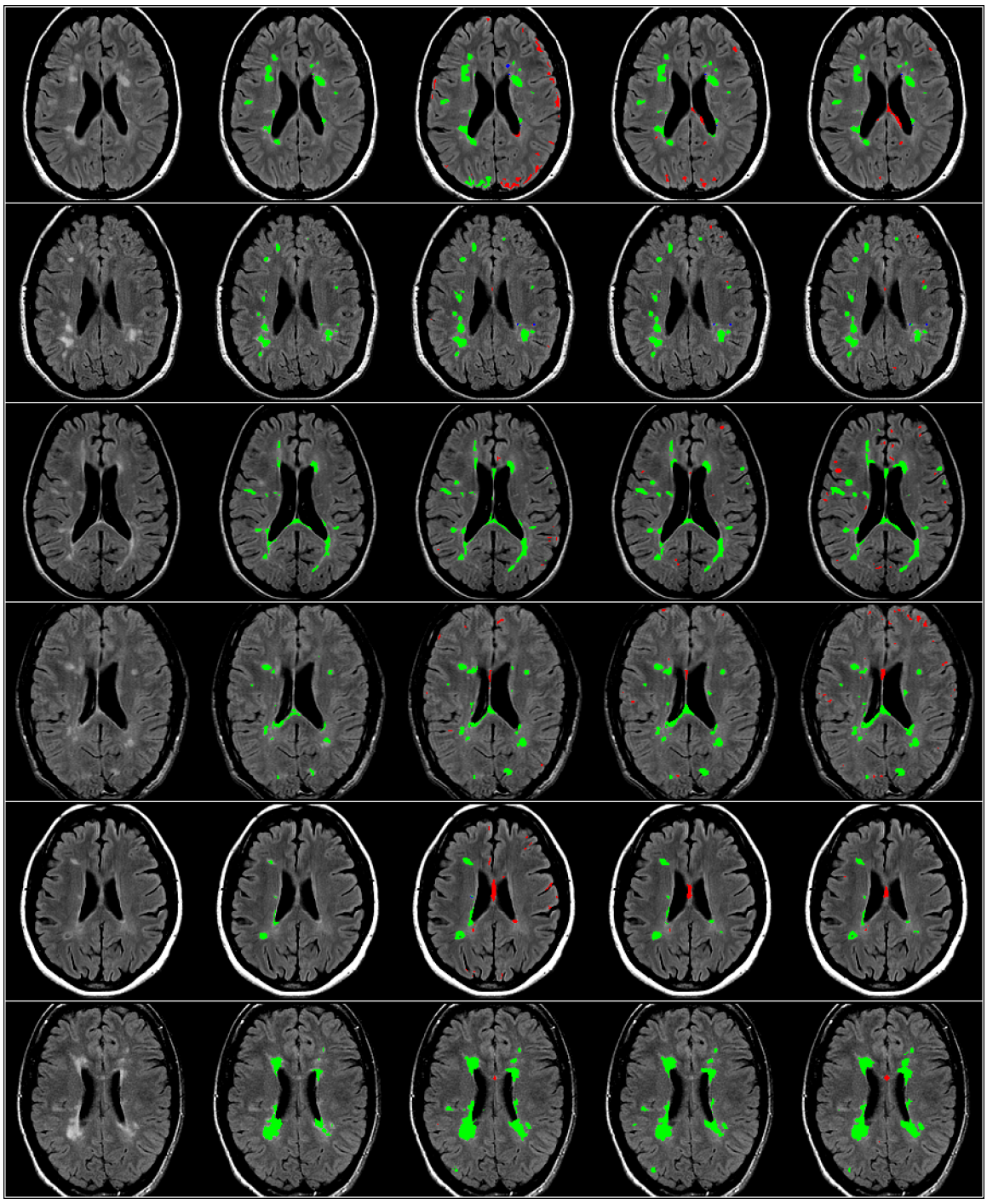


Figure 5.7 Qualitative Level-1 Classification Results. This image shows the FLAIR images of 7 different subjects from the Dataset A testing set from top to bottom. Left-to-right: (a) original FLAIR image, (b) manually labeled lesions, (c) results from the 3x3x3 neighbourhood classifier using architecture #p3c5a, (d) results from the 9x9x9 neighbourhood classifier using architecture #p9c1, and (e) results from the 23x23x23 neighbourhood classifier using architecture #p23c6. The colors green, red and blue represent TP, FP and FN respectively.

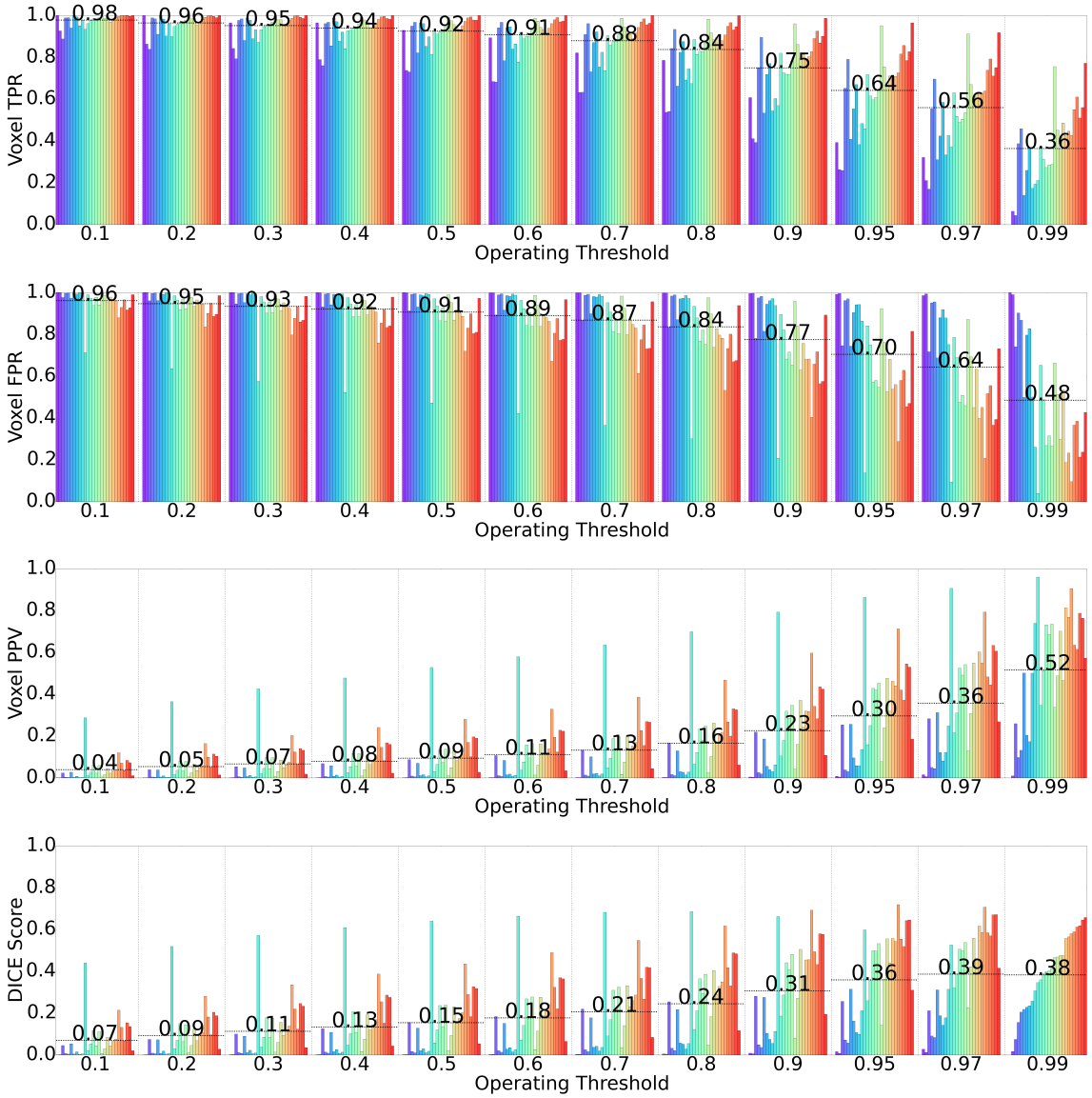


Figure 5.8 3x3x3 Voxel-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) TPR, (b) FPR, (c) PPV, and (d) Dice score. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

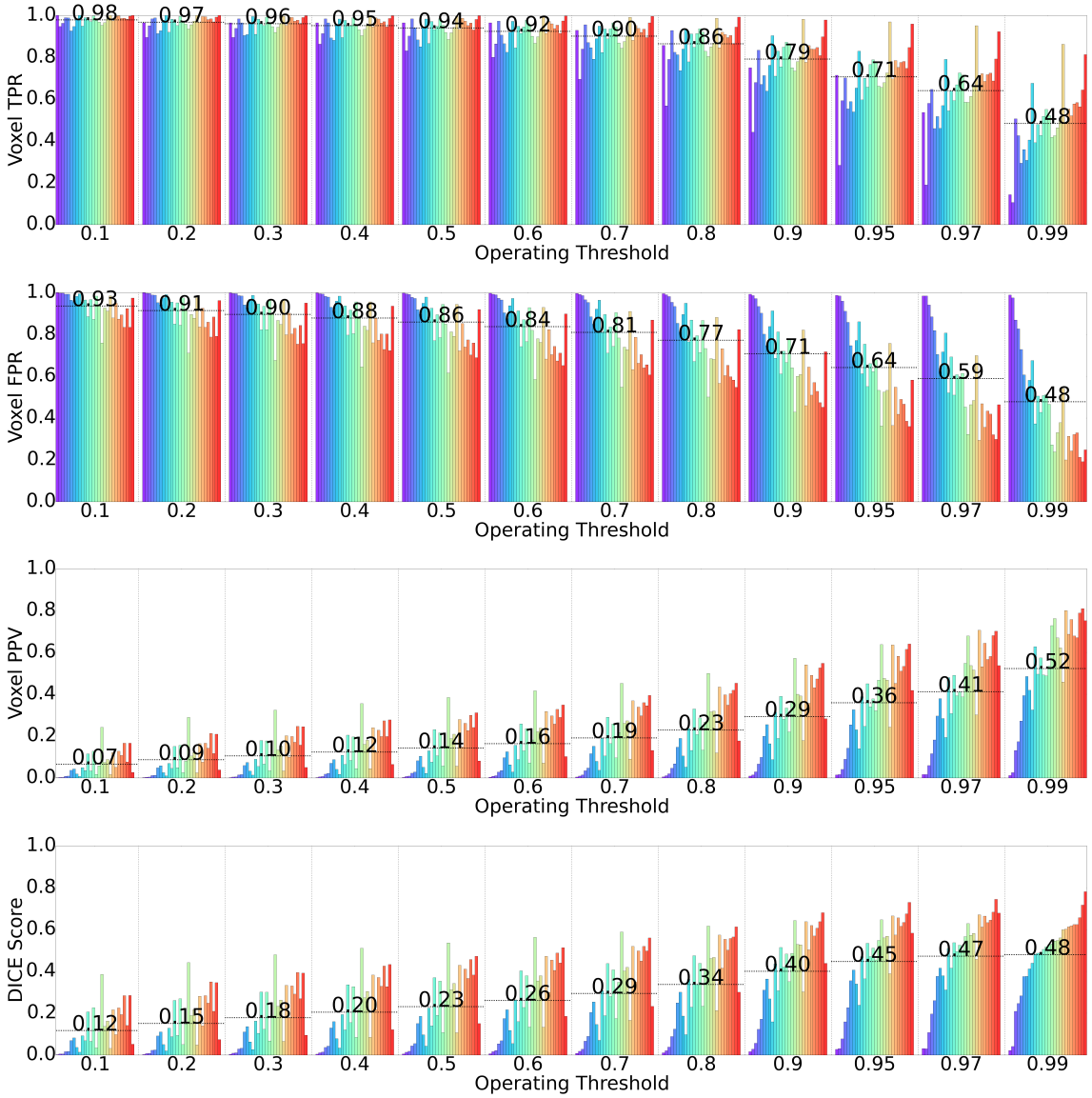


Figure 5.9 9x9x9 Voxel-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) TPR, (b) FPR, (c) PPV, and (d) Dice score. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

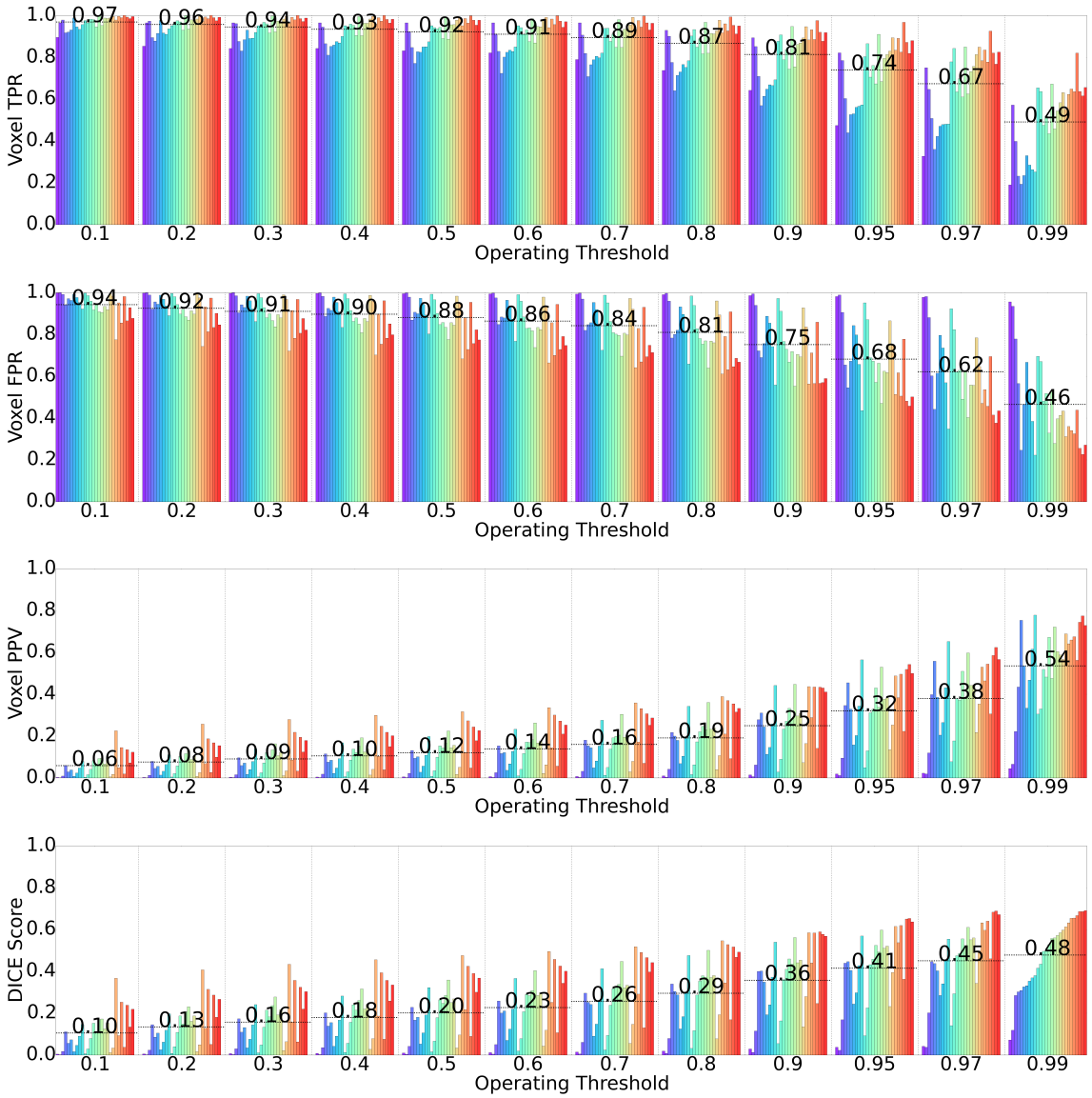


Figure 5.10 23x23x23 Voxel-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) TPR, (b) FPR, (c) PPV, and (d) Dice score. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

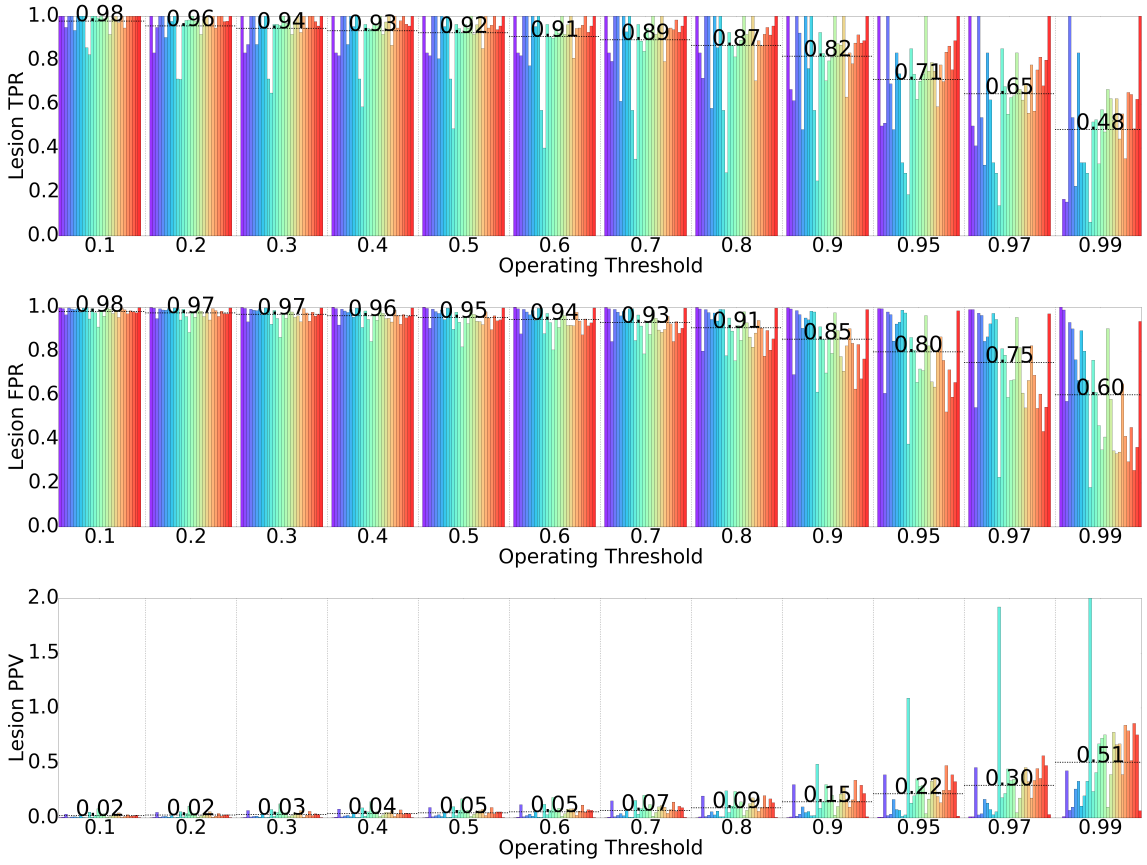


Figure 5.11 3x3x3 Lesion-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) LTPR, (b) LFPR, and (c) LPPV. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

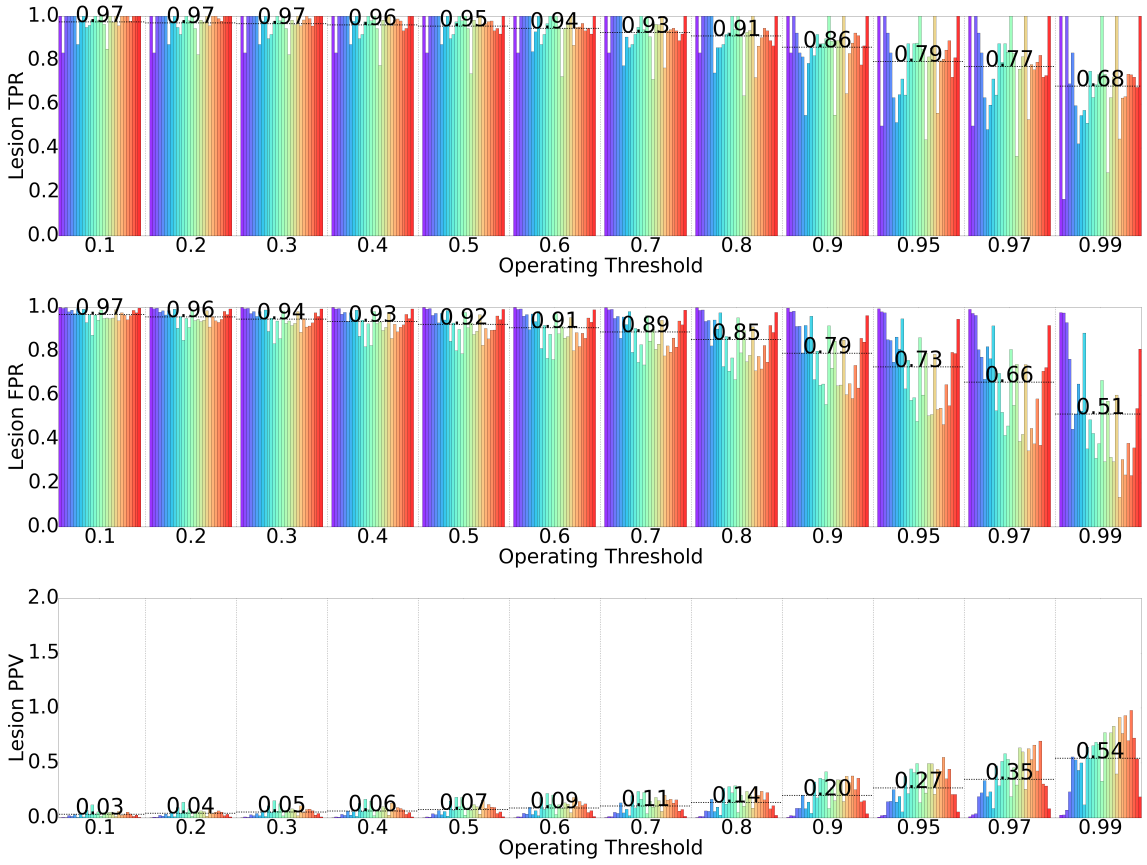


Figure 5.12 9x9x9 Lesion-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) LTPR, (b) LFPR, and (c) LPPV. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

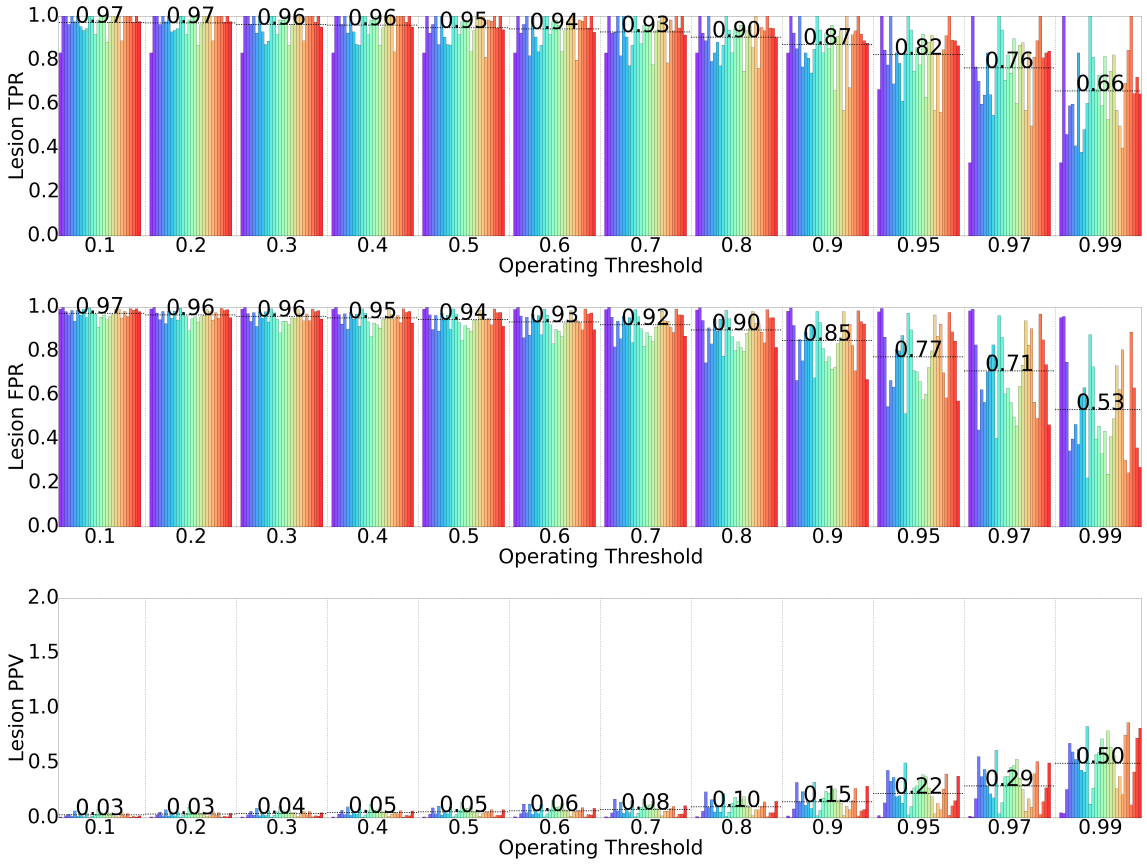


Figure 5.13 23x23x23 Lesion-Wise Level-1 Classification Performance. Each subject is shown using a different color-code in order to show the variability over the 28 testing subjects. Top-to-bottom: (a) LTPR, (b) LFPR, and (c) LPPV. The operating threshold is the probabilistic confidence level of the neural network that was taken as the positive decision threshold. The mean metric value is annotated for each operating threshold.

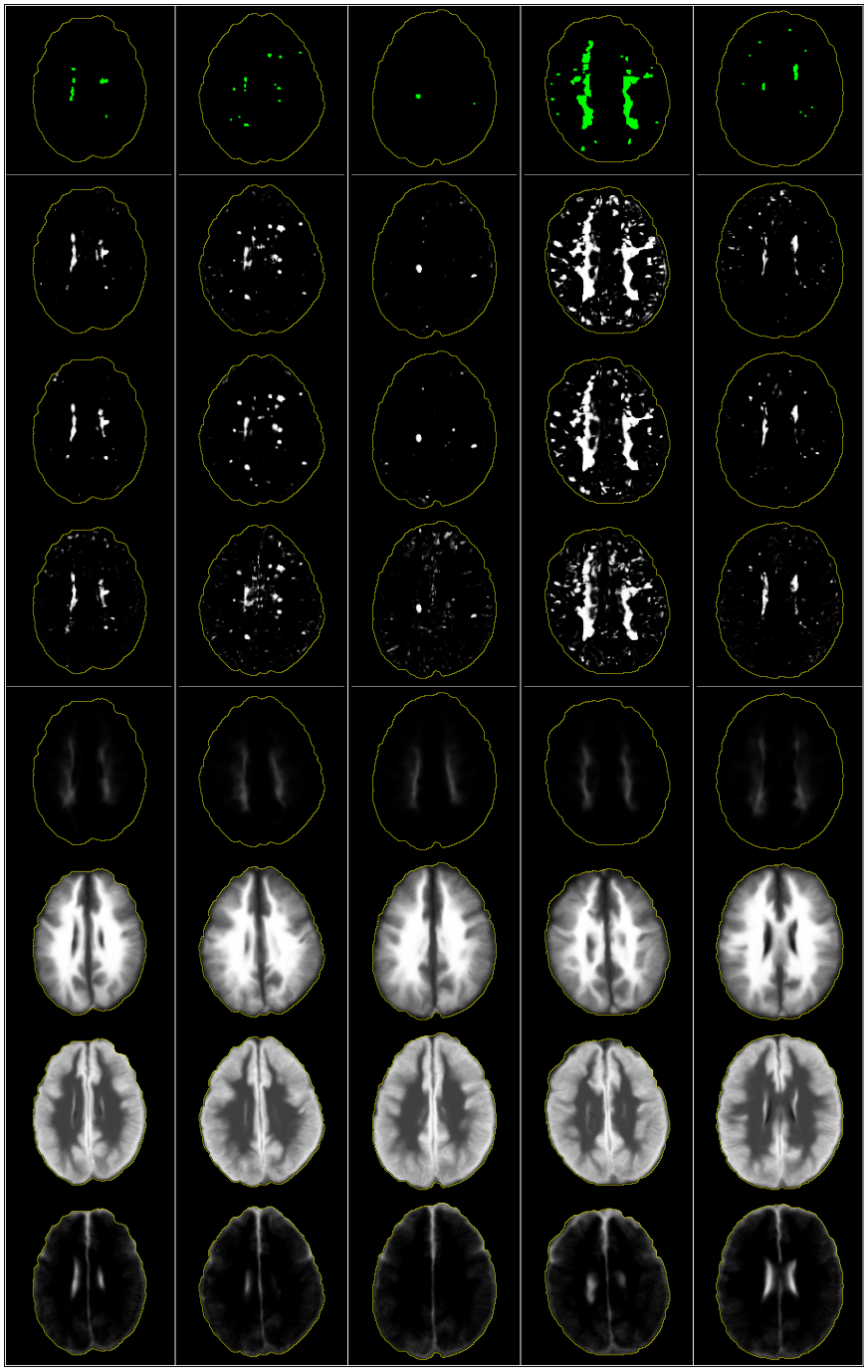


Figure 5.14 Level-2 Inputs. This image shows the Level-2 inputs and ground-truth for 5 different subjects from left to right. Top-to-bottom: binary ground-truth expert labels, probabilistic results from the 27x27x27 CNN in Level-1, probabilistic results from the 9x9x9 CNN in Level-1, probabilistic results from the 3x3x3 classifier in Level-1, probabilistic lesion atlas priors, probabilistic white-matter spatial prior, probabilistic gray-matter spatial prior and probabilistic cerebrospinal fluid spatial prior. The contour of the brain-mask is shown in yellow for convenience.

is designed to make a more informed decision at this stage, which is based on the results of all the three Level-1 classifiers. Several sets of ANN models and training hyperparameters have been used for the experiments, and a selection of 8 configurations along with their results results are shown in Table 5.2. The differences between the configurations are the amounts and types of layers, the learning rate, momentum and batch size while trained using the same framework that was implemented for Level-1.

Exp.	l2ann1	l2ann2	l2ann3	l2ann4	l2ann5	l2ann6	l2ann7	l2ann8
Batch	64	64	64	64	9522	95222	95222	95222
L1	ReLU 1024 N	ReLU 7 N	ReLU 8192 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N
L2	ReLU 1024 N	ReLU 7 N	ReLU 8192 N	Dropout 0.5	Dropout 0.5	Dropout 0.5	Dropout 0.5	Dropout 0.5
L3	Softmax 2 N	Softmax 2 N	Softmax 2 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N	ReLU 1024 N
L4				Dropout 0.5	Dropout 0.5	Dropout 0.5	Dropout 0.5	Dropout 0.5
L5				Softmax 2 N	Softmax 2 N	Softmax 2 N	Softmax 2 N	Softmax 2 N
LR	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.001
Mom.	0.9	0.9	0.9	0.9	0.1	0.9	0.1	0.1
Mis.	4.25%	4.08%	4.27%	4.25%	**4.01%**	**4.01%**	4.02%	4.34%

Table 5.2 Level-2 Training Performance Benchmark. The results reflect the misclassification rate for each of the architectures over balanced datasets that were used for validation in the experiments. Exp. = Experiment ID, Batch = examples per training batch, Lx = Layer x, LR = Learning Rate, Mom. = Momentum, Mis. = Misclassification Rate.

The best performing architectures used large amounts of hidden neurons, however, their results are comparable to those obtained using a significantly smaller model capacity in architecture #l2ann2 to obtain 4.08% misclassification rate. Comparing architectures #l2ann2, #l2ann1 and #l2ann3 with 7, 1024 and 8192 neurons per each of their two hidden layers respectively, the better performance was obtained by the model with the smaller capacity of 7 neurons. On the other hand, adding Dropout layers with 50% probabilities reduced the misclassification rate of 1024 neurons per layer from 4.25% to 4.01%. Another important factor that affected the final training performance is the amount of samples used per batch during the gradient descent. The best results were obtained using example amounts in the orders of 10,000 and 100,000 per batch in architectures #l2ann5 and #l2ann6 respectively. The validation results at this level improved from 4.89%, 4.37% and 4.56% misclassification rates

of the three individual Level-1 models to 4.01% for the balanced validation misclassification rate of Level-2, as shown in Table 5.3.

Level	Model	Misclass. Rate
L1	CNN1-1 (3x3x3)	4.89%
	CNN1-2 (9x9x9)	4.37%
	CNN1-3 (23x23x23)	4.56%
L2	ANN2	**4.01**%

Table 5.3 Level-2 Validation Performance. The results reflect the misclassification rate for each of the architectures over balanced datasets that were used for validation in the experiments.

5.7 Level-3 Results

The last level of the method, Level-3, is intended to eliminate false-positive lesion detections from the previous layer, and consequentially, its training database was generated using predictions from the trained Level-2 classifier. This is the first time that the binary classification results form lesions, and are classified as such. In the previous stages, every voxel was considered separately, and in Level-3, these voxels are merged into full lesion based on the adjacency model described in Section 5.4. The training data was balanced in an lesion-wise manner, which means that 50% of the samples are true-positive lesions detections, and the other 50% are false-positives. At this stage, it is not expected to recover a higher sensitivity score, however, it was expected that given high-quality ground-truth labeling the CNN will learn to distinguish between false-positives and true-positive on an object-basis rather than at the voxel-level. Example inputs to Level-3 are shown in Figure 5.15.

The data for this stage set was generated using 60 patients for training, 24 for validation and 20 for testing from Dataset A, and the total amounts of samples used for the training of the CNNs were 5750 and 1000 for the balanced training and validation sets, respectively. Several different architectures were experimented with for the false-positive elimination task, and their results are shown in Table 5.4. The lowest balanced-dataset misclassification rate of 13.0% was obtained by architecture #l3c2, yet, a comparable rate of 13.1% was obtained by model #l3c4 with less neurons-capacity. The first model uses two convolutional layers of 128 neurons each, while the second one uses only 32, in addition to two fully-connected ReLU layers with 1024 units that were replaced with 64 units each. The deepest model, #l3c3, reached only 17.7% misclassification rate, but due to the dropout it is possible that

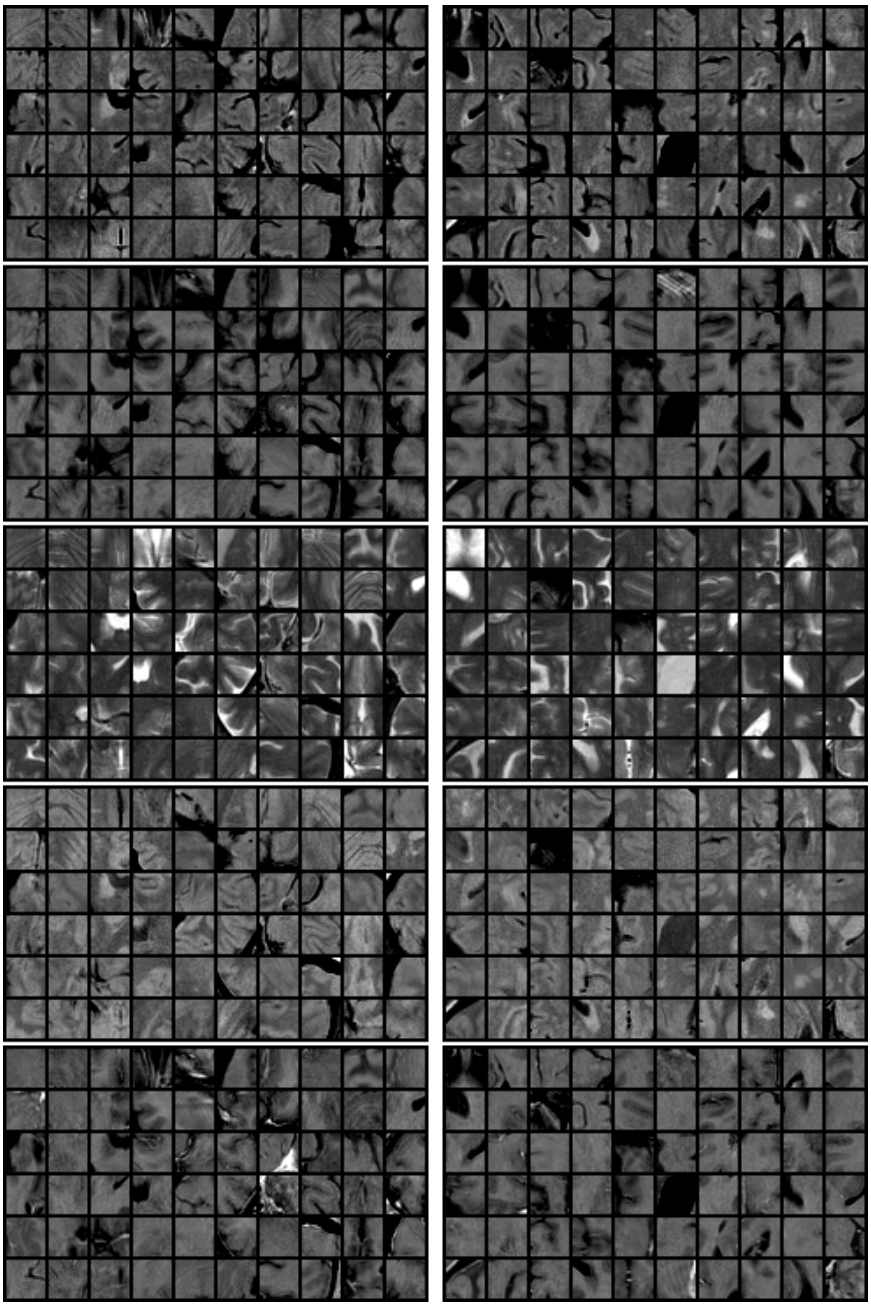


Figure 5.15 Layer-3 Examples. The images are the middle axial slice of the MRI examples used to train the first level of convolutional neural network classifiers. The patches shown are of Level-2 false-positive examples (non-lesions) on the left-hand side and true-positive examples (lesions) on the right-hand side. The image shown different 60 different examples from each class, grouped from top to bottom: FLAIR, T1-weighted, T2-weighted, PD, and T1-post-contrast examples.

longer, time-consuming training could have been beneficial to its final score. Architecture #l3c5 experimented with an increased convolutional kernel size for the first layer of the CNN, followed by a smaller kernel in the consequent layer and yielded 14.0% misclassification rate, which could be attributed to the structure of the model in this case. The reason is the previous model with similar layers, but different kernel sizes obtained a lower misclassification rate using the same optimization method. As we can see from configuration #l3c6, a shallower model containing only a single convolutional layer achieved a lower score of 15.1%.

Exp.	l3c2	l3c3	l3c4	l3c5	l3c6
L1	Conv. 3x3x3 128 N	Dropout 0.8	Conv. 3x3x3 32 N	Conv. 5x5x5 32 N	Conv. 5x5x5 32 N
L2	Pooling 2x2x2	Conv. 3x3x3 128 N	Pooling 2x2x2	Pooling 2x2x2	Pooling 2x2x2
L3	Conv. 3x3x3 128 N	Pooling 2x2x2	Conv. 3x3x3 32 N	Conv. 2x2x2 32 N	ReLU 64 N
L4	Pooling 2x2x2	Dropout 0.5	Pooling 2x2x2	Pooling 2x2x2	ReLU 64 N
L5	ReLU 1024 N	Conv. 3x3x3 128 N	ReLU 64 N	ReLU 64 N	Softmax 2 N
L6	ReLU 1024 N	Pooling 2x2x2	ReLU 64 N	ReLU 64 N	
L7	Softmax 2 N	Dropout 0.5	Softmax 2 N	Softmax 2 N	
L8		ReLU 1024 N			
L9		Dropout 0.5			
L10		ReLU 1024 N			
L11		Dropout 0.5			
L12		Softmax 2 N			
LR	0.01	0.01	0.01	0.01	0.01
Mom.	0.1	0.1	0.1	0.1	0.1
Mis.	**13.0%**	17.7%	13.1%	14.0%	15.1%

Table 5.4 Level-3 Training Performance Benchmark. The results reflect the misclassification rate for each of the architectures over balanced datasets that were used for validation in the experiments. Exp. = Experiment ID, Batch = examples per training batch, Lx = Layer x, LR = Learning Rate, Mom. = Momentum, Mis. = Misclassification Rate.

5.8 Combining Features in a Random Forest Classifier

The second level of the method is designed to model the joint distribution over the first level classifiers and the spatial priors in order to yield more informed lesion predictions. The first approach used an artificial neural network for that purpose. In this section, a probabilistic random forest classifier, RF2-1 in Figure 4.4, was trained to model the same function. Using only the final classifications from the first level could be limiting in terms of discriminative power. The reason is that given the large sets of features that led each of the classifiers to yield their final decision is much more informative than just a single value between 0 and 1, and when considering several different classifiers, this is similar to combining their complete reasoning process rather than only their final classifications. Therefore, the next logical step was to further combine the last-layer features from the 3 Level-1 classifiers, rather than only using their final predictions. The second approach is to train the random forest classifier, RF2-2 in Figure 4.4, over these features.

The ROC curves for the results of the two approaches are shown in Figure 5.16. For completeness, the ROC curves for all other proposed architectures throughout this thesis are also presented, and were obtained using the identical training, validation and testing sets. The mean ROC curves for the results from all methods are compared in Figure 5.17 and their corresponding Area Under Curves (AUCs) are shown in Table 5.5. As we see in Table 5.5, the area under the curve was the largest for the random forest classifiers, both over predictions (RF2-1) and over CNN features (RF2-2), but they were very close and it is hard to pick a real winner. However, the AUC values are very close between CNN1-2, ANN2, RF2-1 and RF2-2, which indicate that there is no improvement in this measure between the best Level-1 classifier and the final results of Level-2 and Level-3. Another interesting result is the confirmation that a small 3x3x3 neighbourhood was the least discriminative in terms of classification using this metric, yielding lower score of 0.9857, and high subject-wise variability could be observed in Figure 5.16.

Estimating the total lesion load for a particular patient is important for clinical assessment of disease activity. Therefore, we choose to evaluate the suitability of the proposed method for estimating total lesion load. The total lesion load in cubic centimeters is calculated by multiplying the amount of lesion labeled voxels by the physical dimensions of a single voxel, which are 1x1x3 millimeters for both Dataset A and B. A comparison between the ground-truth and its predictions is shown in Figure 5.18. We can that a simple linear regression between the classifier and the labeling yielded fit coefficients of $y = 0.951x + 0.001$ when

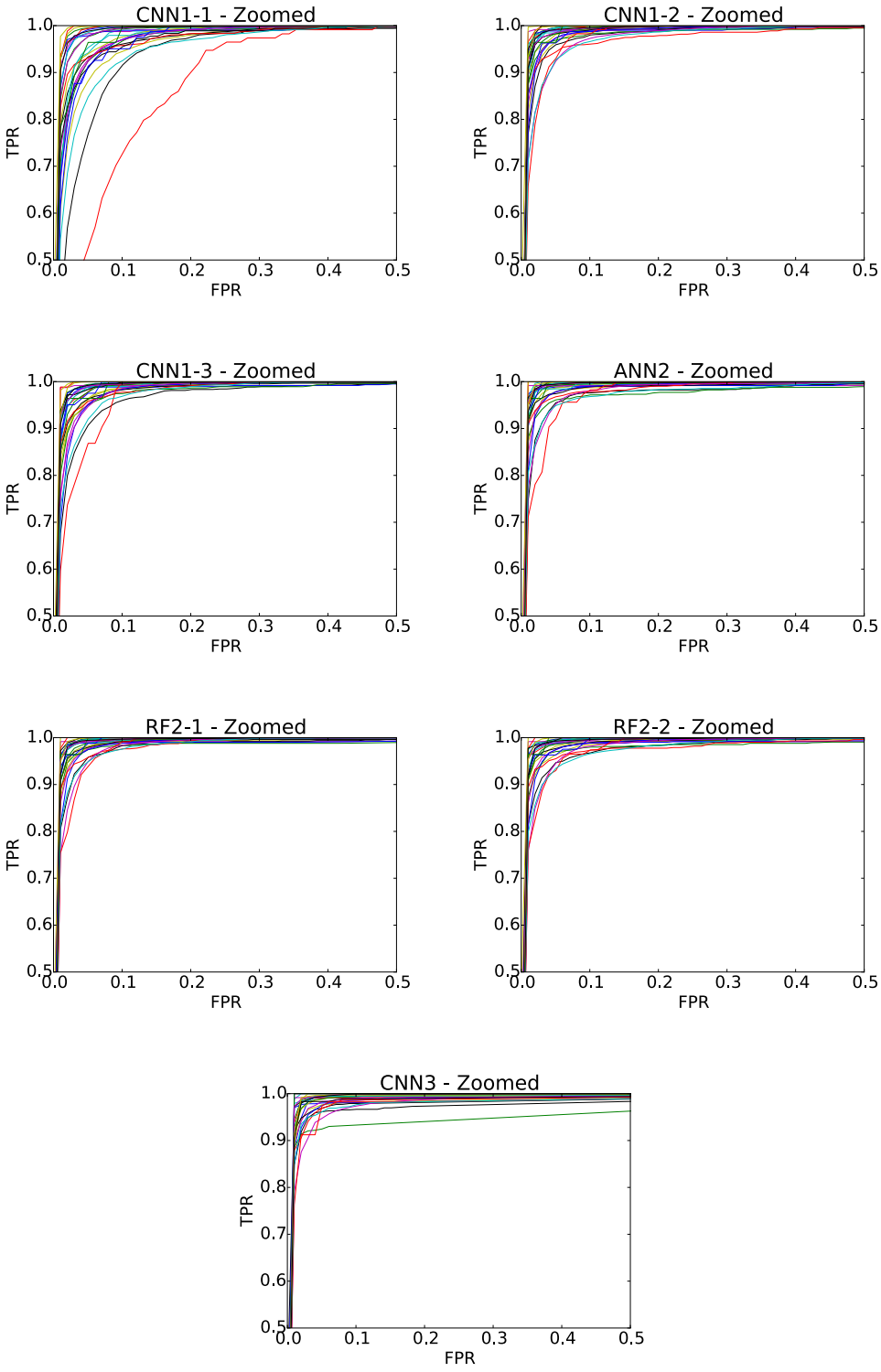


Figure 5.16 Receiver Operating Characteristic. The ROC curves were drawn for all the patients in the testing set, each represented by a different color. The curves were drawn for each of the building blocks of the proposed architectures: CNN1-1, CNN1-2, CNN1-3, ANN2, CNN3, RF2-1 and RF2-2.

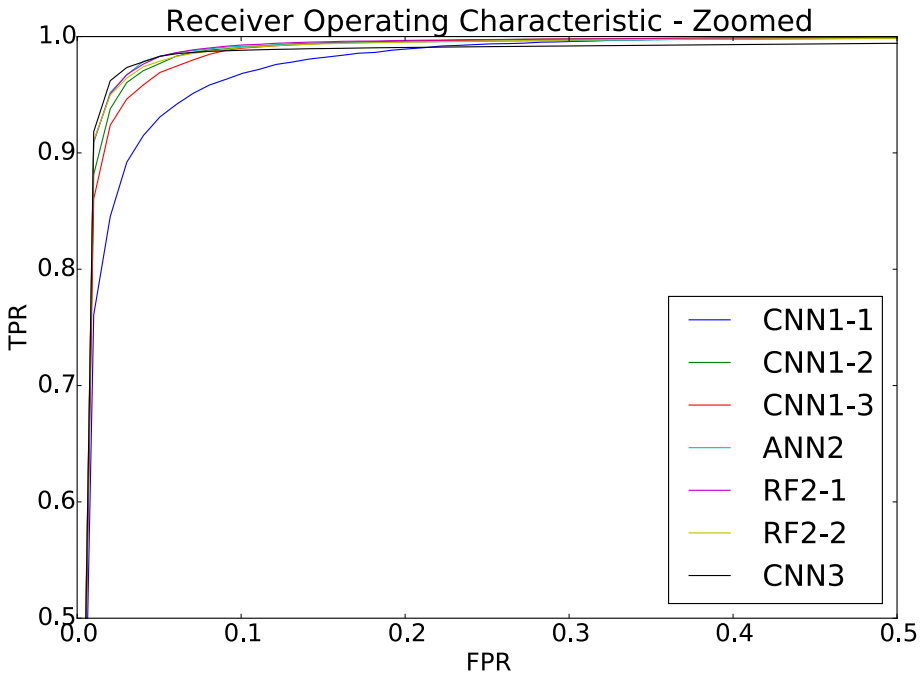


Figure 5.17 Mean Receiver Operating Characteristic. This Figure shows the mean ROC, averaged over all testing patients, in order to compare between the different classifiers at each stage of the method.

Model	AUC	Description
CNN1-1	0.9857	Level-1 Voxel-Intensity Model
CNN1-2	0.9935	Level-1 Close-Neighbourhood Model
CNN1-3	0.9923	Level-1 Larger-Neighbourhood Model
ANN2	0.9937	Level-2 Artificial Neural Network Model
RF2-1	0.9943	Level-2 Random Forest using Level-1 Outputs
RF2-2	0.9937	Level-2 Random Forest using Level-1 Features
CNN3	0.9907	Level-3 False-Positive Elimination

Table 5.5 Area Under the ROC Curves (AUC). The results reflect the mean area under the ROC curve for each of the models used in the method, averaged over all testing patients.

minimizing the mean square error (MSE).

5.9 Deep-Learned Features

Prediction using the hidden CNN features directly, using a random forest, was shown to be comparable in accuracy to the other proposed approaches in Tables 5.5 and 5.6. It would be interesting to further explore the extracted features and better understand them. These features were shown to provide discriminative information about lesion labels at each voxel, and were learned in 3 different scales in architectures CNN1-1, CNN1-2 and CNN1-3. In order to further explore this idea, one can visualize these features over a randomly selected set of patients assess their viability. Figure 5.19 presents 111 out of the 4096 available CNN-learned features, along with the original image. For convenience, the features were overlayed on the original FLAIR modality, while showing them only in regions that carry a non-zero activation value. Features seem to carry information that could be useful for other tasks in addition to lesion-segmentation, such as brain tumor and stroke lesion segmentation, and therefore open the opportunity to apply either transfer learning [63], or use other classifiers, such as Bayesian approaches, while employing the results of the CNNs only as features.

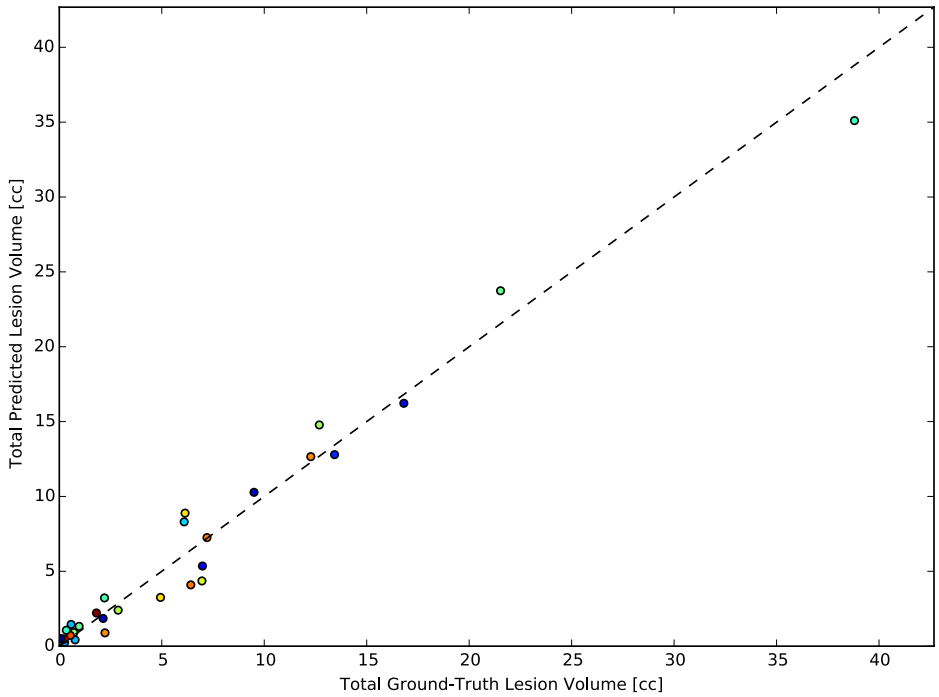


Figure 5.18 Total Lesion Load Comparison. The purpose of this comparison is to evaluate whether the total volume of the segmented lesions correspond to the total volume of ground truth lesions. Each of the testing subjects is presented as a point over the graph of ground-truth total lesion load versus total predicted lesion volume in cubic centimeters (cc). The total lesion load is calculated by multiplying the amount of lesion labeled voxels by the physical dimensions of a single voxel, which are 1x1x3 millimeters. For convenience, the equivalence $y = x$ was also added as a dashed line over the scattering. A linear fit over the curve, minimizing the mean square error (MSE) yielded fit coefficients of $y = 0.951x + 0.001$.

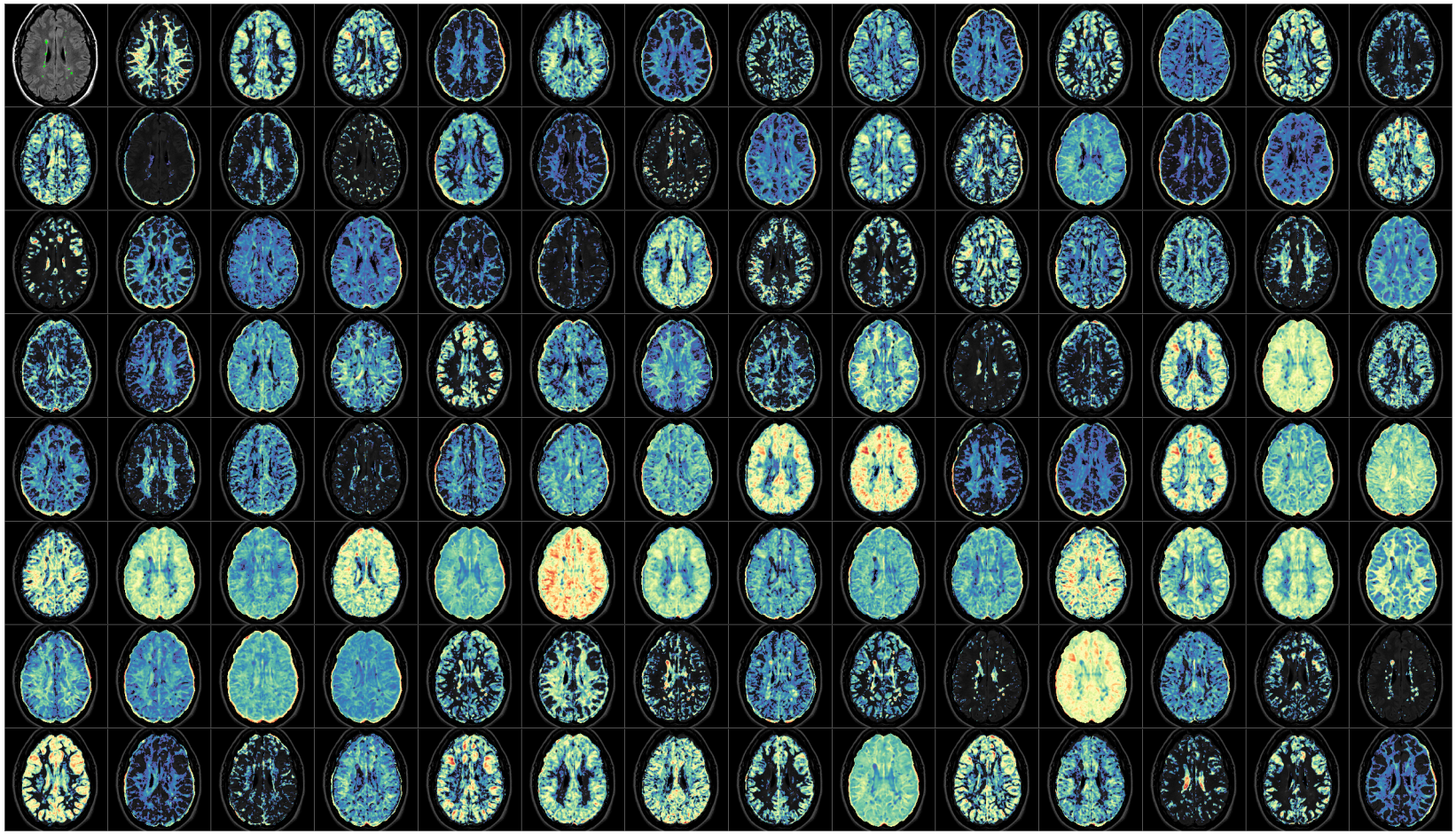


Figure 5.19 CNN Features 1. This figure shows examples of activation levels of a sample set of features from the 4096 features extracted by the last hidden layer activations of the three Level-1 convolutional neural networks. Top-left is the original FLAIR image with lesion labeling in green, followed by 111 of its CNN features overlayed over the FLAIR modality. Blue and red signify the strength of the hidden activation from low to high, respectively.

In Figure 5.20, the degree to which the features are consistent of a set of different patient images is examined. The original FLAIR images are shown at the top along with their ground-truth labeling, and the response from the same 7 feature descriptors are shown below each of them for 14 different patient images. These features seem to carry information that qualitatively correspond to similar regions in different images, and are obtaining comparable values. For example, the fourth row corresponds to a feature that captures the circumference of the brain, within the brain-mask, and maintains a similar response over all images. The second row corresponds to a feature that activates at the white matter, but not in the presence of a lesion. Those observations are qualitative only, since those features are learned from the data and not hand-crafted, and it is unfeasible to predict their response to new examples, however, these features are trained to the perform the task of providing a linear separation between lesion and non-lesions as shown in Equation 3.6.

The next step was to examine the ability to separate between classes given the CNN features. Since visualizing 4,096 features-maps over the each slice would not be human-interpretable or readable, a different approach was suggested. A balanced set of 10,000 examples were taken from the database, while 5,000 are labeled as lesions and the other 5,000 are non-lesion examples. In Figure 5.21, the 4,096 features for each example were flattened into a column vector and drawn as a vertical line. The left-hand side of the image was filled by the 5,000 lesion examples, and the right-hand side was filled by the other 5,000 non-lesion vertical lines. We can see two forms of descriptors that resemble a bar-code for lesions on the left, and a bar-code for non-lesion on the right. Recalling that those are the features that are used by the classifier to distinguish between lesions and non-lesion, qualitatively, it appears that the task is well-defined by the presented features for most examples. Another interesting discovery is that the top quarter (first 1,024 of 4,096 features) in the figure is taken from the 3x3x3 classifier, in which we can see that its features are prone to more noise than the ones obtained by the more stable 9x9x9 and 23x23x23 neighbourhood classifiers. These also explain why the predictions provided by the 3x3x3 yielded lower scores and smaller area under the ROC curve. The respective average variance values for the 3x3x3, 9x9x9 and 23x23x23 features are: 0.03663 ± 0.00195, 0.00347 ± 0.00005 and 0.00718 ± 0.00020. The average entropy values for the same set of classifiers are: 10.75 ± 0.80, 11.06 ± 4.55 and 11.34 ± 1.80 respectively. This shows that the average entropies are all within the same order of magnitude, and the average variance of the 3x3x3 features is one order of magnitude higher than the 9x9x9 and 23x23x23 features. This supports the qualitative observation that there is noise in the features obtained by the 3x3x3 CNN.

In order to provide the reader with practical understanding of how the internal feature-extraction process takes place within the convolutional neural networks used in this thesis, the first convolutional layer kernels from CNN1-2 are shown in Figure 5.22. These kernels are used as filters and convolved with the input image to extract the first level of internal hidden representations, which are the feature maps, and are further convolved with the next layer of kernels up to the last layer of the CNN. Although it is hard to gain additional insight about the features by visual inspection of these kernels, we can still see that the low-level features capture edges in several directions and fine textures. Hierarchical pooling of decompositions of the 5 MRI modalities using these kernels yield the final classification results reported throughout this thesis, and they are presented here for completeness.

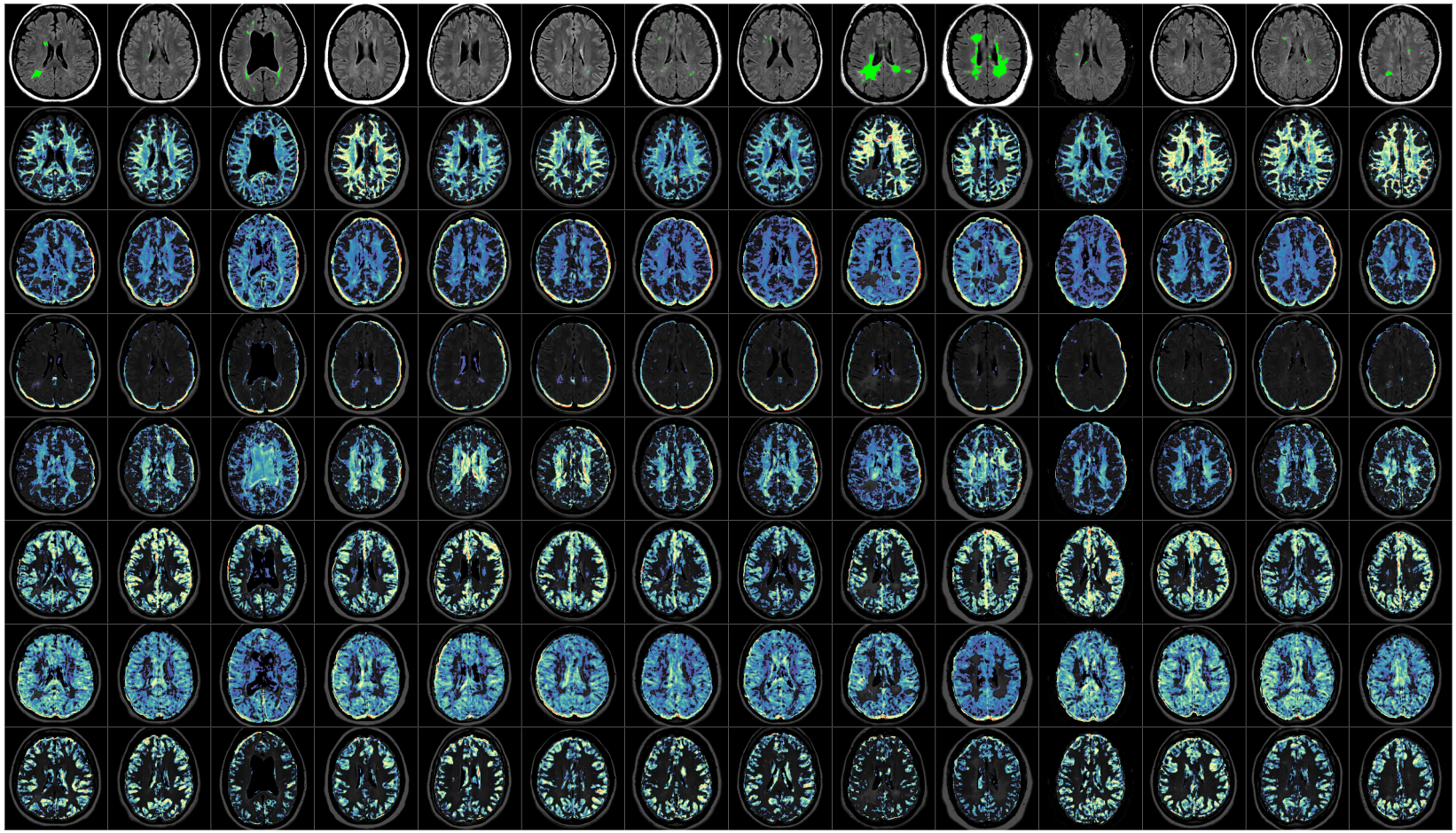


Figure 5.20 CNN Features 2. A selection of 7 CNN features are shown over 14 different patients from the dataset. The top row is the original FLAIR image with lesion labeling in green, followed by the corresponding CNN features overlayed over the FLAIR modality. Blue and red signify the strength of the hidden activation from low to high, respectively. Notice that the same features (rows) correspond to similarly appearing structures for different patients (columns). Some features seem to describe very distinct structures in the brains such as healthy white matter structures in the second row, CSF that was not filtered out in the brain extraction in the fourth row, gray matter in the last row, etc. Recall that these features were learned automatically by the CNN.

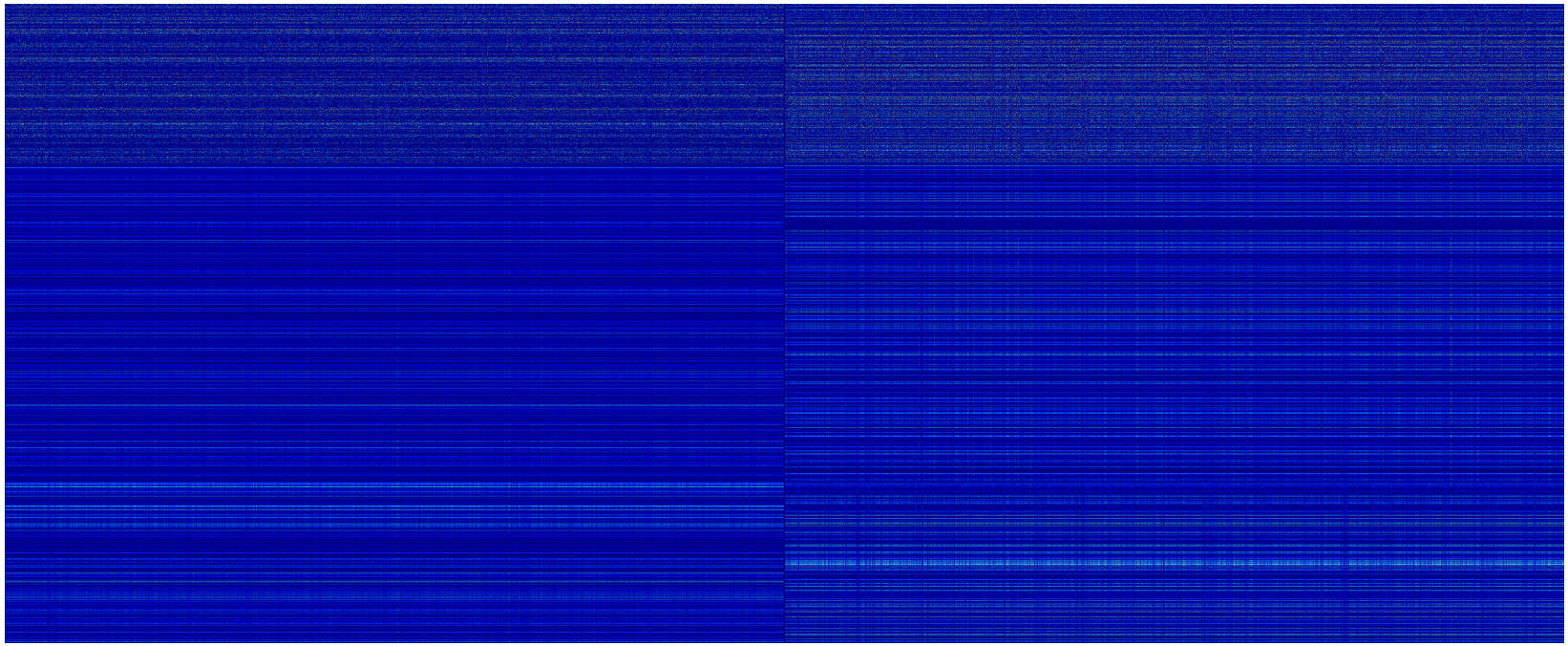


Figure 5.21 Lesions vs. Non-Lesions. This image shows the flattened hidden feature vectors of lesion- and non-lesion samples from the dataset. Each vertical line corresponds to the flattened vector of CNN features for a single example, of length 4096 features. The left-hand side of the figure corresponds to lesion examples, and the right-hand side corresponds to non-lesions. The top quarter corresponds to the 1024 features from CNN1-1, the bottom quarter to the 1024 features from CNN1-3, and the middle to the 2048 features from CNN1-2. Notice that the features of lesion examples on the left-hand side appear similar to each others, and very different from non-lesion examples on the right-hand side. A careful observation at the features reveals another interesting result: the features extracted by the smallest neighbourhood model 3x3x3 at the top quarter appears to be more prone to noise than those produced by the larger neighbourhood models below. The respective average variance values for the 3x3x3, 9x9x9 and 23x23x23 classifiers are: 0.03663 ± 0.00195, 0.00347 ± 0.00005 and 0.00718 ± 0.00020. The average entropy values for the same set of classifiers are: 10.75 ± 0.80, 11.06 ± 4.55 and 11.34 ± 1.80 respectively. This shows that the average entropies are all within the same order of magnitude, and the average variance of the 3x3x3 features is one order of magnitude higher than the 9x9x9 and 23x23x23 features. This supports the qualitative observation that there is noise in the features obtained by the 3x3x3 CNN.

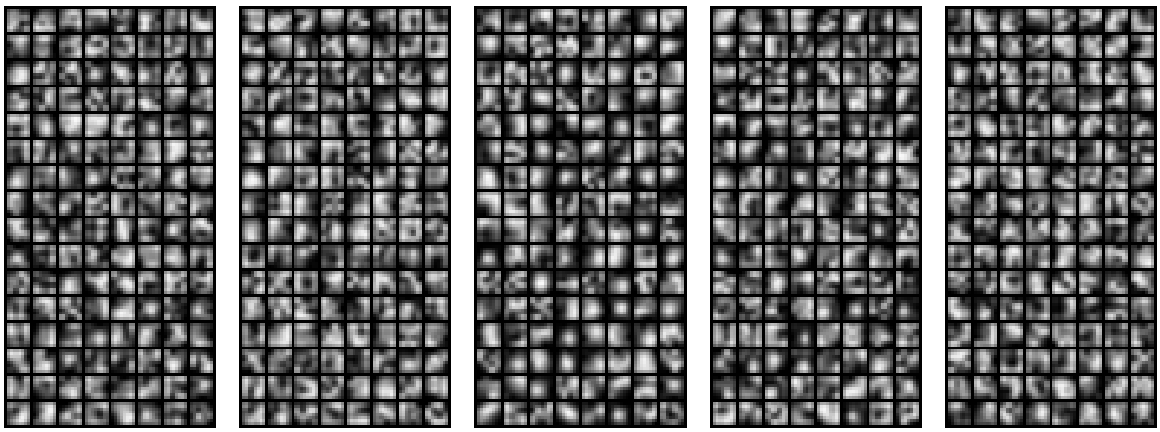


Figure 5.22 First Layer Convolution Kernels. The middle axial slices from the 128 kernels used by the first convolutional layer of CNN1-2 are shown in this figure. These kernels are used as filters and convolved with the input image to extract the first level of internal hidden representations, which are the feature maps, and are further convolved with the next layer of kernels up to the last layer of the CNN. We can see that these low-level features capture edges in several directions and fine textures. Left-to-right: FLAIR, T1-weighted, T2-weighted, PD, and T1-post-contrast kernels.

5.10 Results on Dataset B

In this section, we explore the working of the method on a completely different, and much larger clinical trial dataset. The selected architectures were carried over from the first dataset, including their layers and optimization hyperparameters, and were completely retrained using Dataset B only. The dataset contains a total of 1,063 patients, from which 835 were used for training, 200 for validation and 28 for testing.

The performance of the method over Dataset A and Dataset B were evaluated and calculated over their respective testing sets. For Dataset A, all the different architectures presented in the previous section were trained and evaluated, and for Dataset B Level-1 classifiers CNN1-1, CNN1-2 and CNN1-3 were trained as feature extractors for the random forest RF2-2. In order to fairly compare the results to other recent methods from the literature, a method that uses both local and regional information was selected. The method, published by Harmouche et al.[37] in 2015, is a fully automatic MS segmentation technique that builds regional likelihood models for tissues and incorporates neighbourhood information using a Markov Random Field (MRF). Also, a re-implementation of their code was available for my lab, and therefore could be tested while using identical training and testing folds over the

proprietary Dataset A. The results are shown in Table 5.6, along with the results obtained using the re-implementation of Harmouche et al.[37].

As we can see, for both datasets A and B, the Dice score was higher at the second level using RF2-2 that models the joint distribution of the first level classifiers. For Dataset A, the Dice score resulting from the additional processing stage reached 0.54, and started from scores of 0.38, 0.48 and 0.48 in the previous level as inputs. For Dataset B, the Dice score of the second level (RF2-2) was 0.59, while the previous level scored 0.36, 0.52 and 0.55 for CNN1-1, CNN1-2 and CNN1-3, respectively. The highest LPPV of 0.76 was obtained using architecture RF2-1 over Dataset A, however, the LTPR of this architecture was lower (0.53) than 2 of the 3 first level classifiers over the same dataset that scored 0.49, 0.68 and 0.66 for CNN1-1, CNN1-2 and CNN1-3, respectively. Recall the LTPR refers to the ratio between true positives and total ground truth lesion count, which means that a lower score in this criteria is an indicator for missing lesion detections. Comparing the results to Harmouche et al.[37] over Dataset A, the Random Forest architecture RF2-2, obtained respective higher Dice scores of 0.54, comparing to 0.51. This architecture also obtained higher LTPR of 0.64 comparing to 0.56, which means that the ratio between true positive lesion detections to total ground-truth lesions count was higher, but lower LPPV.

Dataset	Model	Dice	LTPR	LPPV
A	Harmouche et al.[37]	0.51	0.56	0.63
A	CNN1-1	0.38	0.49	0.49
	CNN1-2	0.48	**0.68**	0.52
	CNN1-3	0.48	0.66	0.48
	ANN2	0.45	0.51	0.61
	CNN3	0.48	0.51	0.69
	RF2-1	0.52	0.53	**0.76**
	RF2-2	**0.54**	**0.64**	0.59
B	CNN1-1	0.36	0.56	0.38
	CNN1-2	0.52	0.55	**0.61**
	CNN1-3	0.55	**0.63**	0.56
	RF2-2	**0.59**	**0.64**	0.53

Table 5.6 Models Performance. The proposed models results are shown both as lesion-wise metrics and Dice scores.

In order to better understand the quality and characteristics of the predictions, the lesion-wise metrics were grouped into lesion volume groups of 0-10, 11-50 and 50+ voxels. It is important to divide the results in terms of size, as it is much easier to detect larger lesions

than small ones, and many frameworks can achieve similarly high accuracy for large lesion segmentation. The results are shown in Table 5.7. As we can see, the method performs better over the larger lesions of 11 and up, obtaining lesion true-positive rates (LTPR) of more than 0.86.

Dataset	Lesion Volumes	LTPR	LPPV
A	0-10	0.34	0.59
	11-50	0.86	0.68
	50+	0.98	0.76
B	0-10	0.36	0.3
	11-50	0.87	0.65
	50+	1.00	1.00

Table 5.7 Performance by Lesion Volumes. The lesion-wise metrics, LTPR and LPPV, are grouped by lesion volumes in voxels.

The qualitative results, based on the extracted CNN features are shown in Figure 5.23. In order to get further assessment of the quality of the delineation itself, Figure 5.24 compares between the ground-truth and the result of the proposed method.

5.11 Discussion and Conclusions

Throughout this chapter, 32 different model configurations for 4 different architectures were presented and compared to the re-implementation results of a recently published method [37]. The architectures were configured and the favourable models were selected using Dataset A only. Once they were set, they were used as is for Dataset B, learning only their parameters without any additional dataset-aware fine tuning. The reason for this process was to assure that the proposed method does not overfit to the dataset used to configure it. We have tested 17 different layer- and optimization-hyperparameters configurations for the first level, which was common to all the 4 proposed architectures, and as shown in the results in Table 5.1, conclude that different configurations obtain different scores. This shows the importance of the trial and error process over the layer-structure and optimization of the convolutional neural networks in order to get higher correct classification rates. Interestingly, the two architectures that used Random Forests, RF2-1 and RF2-2, obtained higher Dice scores (0.52 and 0.54) and LTPR (0.53 and 0.64) than the architectures that used the artificial neural network ANN2 (Dice 0.45, LTPR 0.51) or false positive rejection layer CNN3 (Dice 0.48, LTPR 0.51). However, due to the known nature of the manually-labeled datasets used

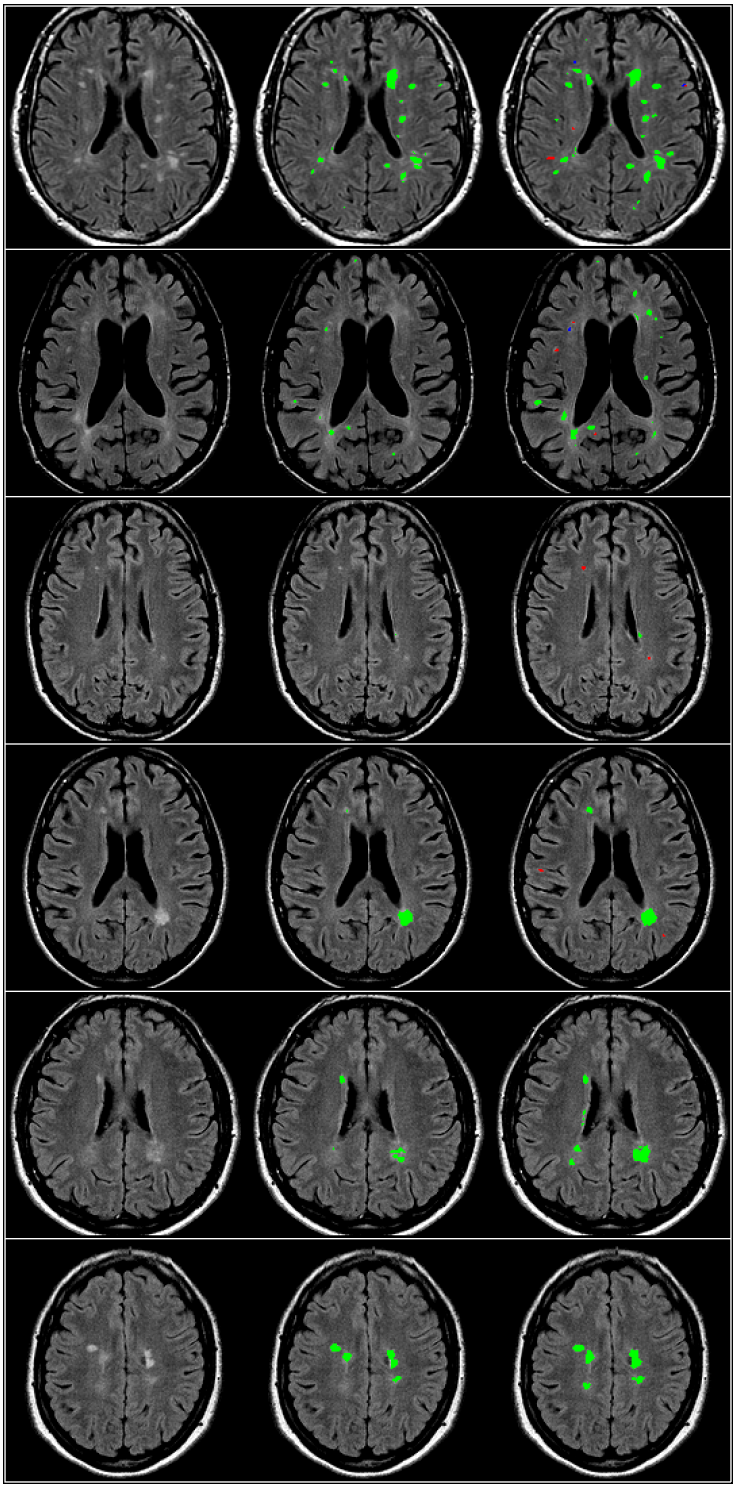


Figure 5.23 Dataset B Qualitative Results. This image shows the FLAIR images of 6 different subjects from the testing set from top to bottom. Left-to-right: (a) original FLAIR image, (b) manually labeled lesions, (c) results from RF2-2, based on the deep-learned features. The colors green, red and blue represent TP, FP and FN respectively.

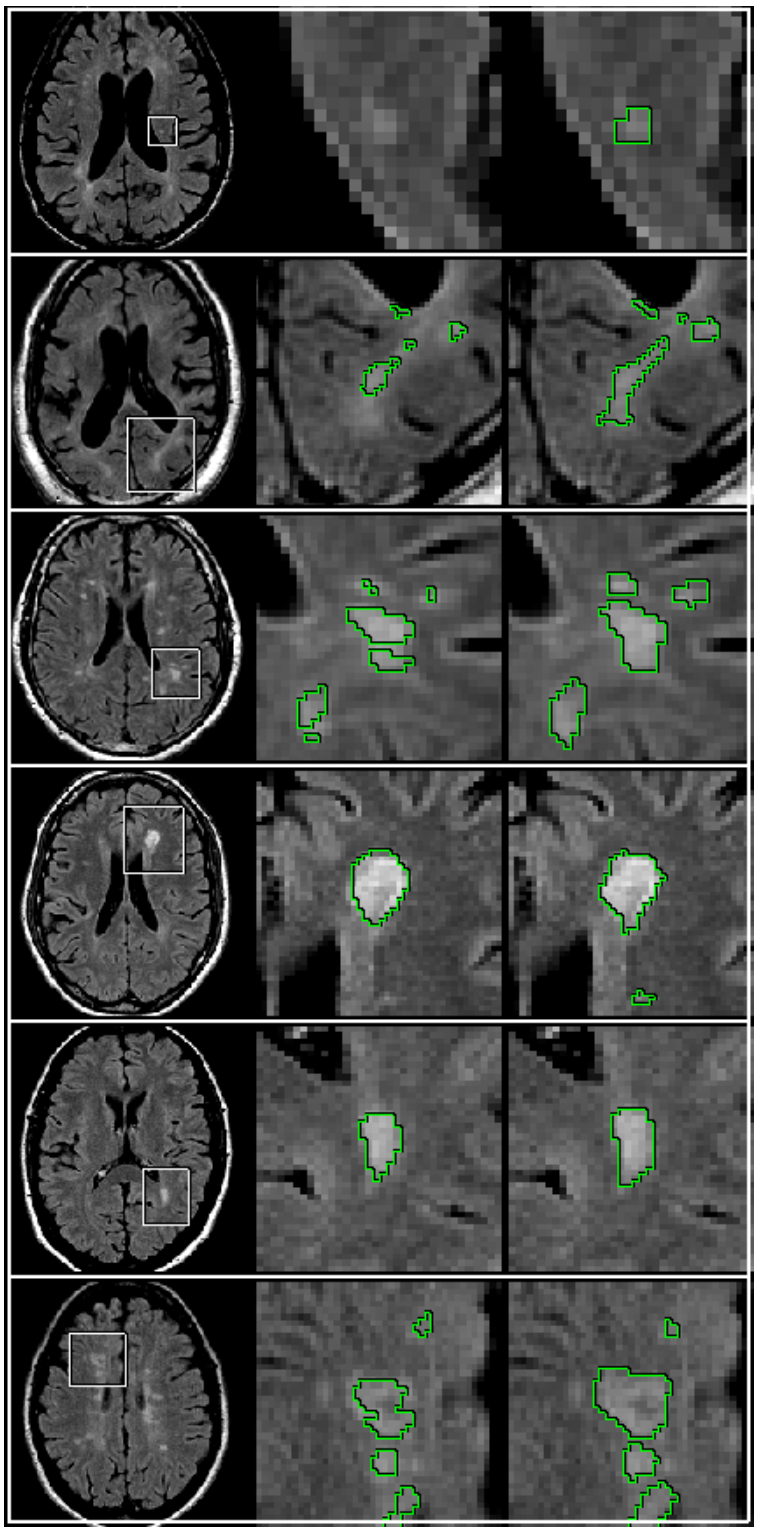


Figure 5.24 Lesion Delineation. This image shows close-ups of lesion true-positive (LTP) delineation over 6 different subjects from top to bottom, comparing ground-truth and the proposed method. Left-to-right: (a) full FLAIR image, close-up regions are marked as white rectangles , (b) ground-truth close up, and (c) proposed method results, based on the deep-learned features. The green contours are the lesion boundaries.

for the experiments, in which lesions objects were occasionally missed from the ground-truth labeling, this false-positive rejection layer could consequentially learn to reject these missed true-positives and, depending on the quality of the labeling, could degrade the overall performance of the CNN3 without any quantitative indicator for this phenomena.

We have also explored the automatically extracted features that were learned by each of the three Level-1 convolutional neural networks, instead of only using their final probabilistic predictions. We have shown that the features for lesion examples appeared similar to each others, and very different from other non-lesion examples. Also, we observed that the features extracted by the smallest neighbourhood model of 3x3x3 appeared to be more prone to noise than those produced by the larger neighbourhoods of 9x9x9 and 27x27x27. Another interesting discovery was that the feature activations corresponded to similarly appearing structures and appeared consistent over different patients. Some of this features seemed to repeatedly describe very distinct structures in the brains such as healthy white matter structures, CSF, gray matter, etc. Exploring these features through the results demonstrated the power of automatic feature extraction using CNNs, and helped to understand the components that are taken into account for generating their outputs for the task.

In the next chapter, I will present the conclusions from this research, the methods that were used and the different proposed architectures. Also, I will further discuss the challenges introduced throughout this research by the Multiple Sclerosis lesion segmentation task, and the methods used to address them. This chapter will continue with further discussion of the outcome of this research and conclude with suggestions for future work.

Chapter 6

Conclusions

This thesis presented a deep-learning approach for MS lesion segmentation, using convolutional neural networks. The main contribution is a new novel adaptation of popular CNN-based deep learning frameworks to the challenging context of MS lesion detection and segmentation, where it is shown to perform well on large multi-center clinical trial datasets, without requiring subjectively defined features. During this work, not only that comparable results to previously published methods were obtain, but also a new set of brain descriptors that could be: (a) generalized and used as features for other tasks, and (b) further investigated to better understand the considerations that lead expert radiologists to their manual labeling protocols. Two major challenges were addressed, the first pertains to the model-parameters selection, which is the key component in optimizing the quality of the results. Many experiments were performed for tuning model parameters such, but not limited to: layers size, amount of layers, convolutions kernels properties, dropout layers, pooling layers size, fully-connected layers and several optimization hyper-parameters. Reporting these results over the diversity of experiments could serve as a guide for future research to avoid re-experimenting with the least successful configurations, and obtain a better starting point. The major challenge was to design the overall architecture, while using convolutional neural networks and other statistical models as its building block, and trying to obtain deep-learned data representations. Several architectures were proposed and compared rigorously over two different large clinical trials proprietary datasets that were made available for this research.

It was demonstrated that even a single convolutional neural network model with the appropriate neighbourhood size and internal parameters could yield comparable results to a previously reported method, while trained and tested over the same data. By adding multi-scale approach, where the components are CNNs, the results were shown to improve in terms

of classification accuracy, lesion-metrics and voxel-wise metrics. Once trained, the model could perform a fully-automated MS lesion segmentation over unseen brain images, taken from different patients, and in other multiple sites. The results from the experiments over the different CNN configurations had shown that there exists a large range of parameters choices that lead to comparable results, rather than just a single optimal configuration, and this could imply that this method could be generalized for other tasks in the future without extensive tweaking of the hyper-parameters.

The task of lesion segmentation was addressed using 3 different high-level architectures, from which several conclusions could be drawn. The first important conclusion is that the extracted hidden CNN features at the last layer could be used by other types of classifiers to obtain comparable results to ones obtained using a neural network. The random forest over the vast amount of features were compared to the ones obtained by each model separately and show similar results. The second conclusion is that the multi-scale approach used for all the proposed architectures improved the overall performance, and yielded better results from each individual scale separately. The third conclusion is that the proposed false-positive elimination CNN (Level-3) degraded the overall performance of the method. This results could be attributed to the quality of the dataset itself, since that architecture was designed to learn how to correct false-positive detections using the trials data, in which many lesions were missed in the ground-truth, and therefore actually learned to eliminate true-positive detections instead. Indeed, a very accurate ground-truth labeling is required for such a post-processing stage, and although it was not beneficial to the results in this thesis, it could be used in the future when higher quality datasets will be available.

Observing the flattened CNN feature vectors for lesion vs. non-lesion examples have been qualitatively and quantitatively shown to provide the classifier with interesting feature descriptors. These features were visually examined and overlayed over a set of brain images, and shown consistency in capturing deep-learned structures within the brain. The feature activations corresponded to similarly appearing structures and appeared consistent over different patients. Some of these features seemed to repeatedly describe very distinct structures in the brains such as healthy white matter structures, CSF, gray matter, etc. Given the fact that these structures provided the reported degree of separation between the lesion and non-lesion classes, it was interesting to overlook and see the components that lead the classifier to obtain its decisions. It was also informative to learn that features obtained from the smallest neighbourhood model were more prone to noise in feature-space, comparing to the larger neighbourhood models, and only a slightly lower level of accuracy was measured in

this case. Furthermore, the total lesion load per patient, which is a useful metric for clinical trials, was evaluated over the proposed architecture and shown a high correlation with the manually-labeled results.

Examples of negative and positive examples from the dataset were presented in to provide the reader with further insight regarding the difficulty of the MS lesion segmentation task. Also, the empirical distributions of lesions over the dataset were plotted in order to show the need for context-information rather than only local-intensity information, and motivate the multi-scale approaches used for the 3 proposed architectures. Interestingly, the method extracted features that do describe some high-level structures of the brain that were presented as the CNN-features. These features could be integrated into different statistical models in future work, such as MRFs that were widely used in the literature. Additionally, it would be interesting to further explore the generalization of these architectures for different classification tasks.

Future work should definitely include expert opinions and analysis regarding the extracted features, or even retraining with a larger amount of input classes, such as brain structures, rather than only pathology detection. This could provide even a richer set of CNN features that could be integrated into many classification or segmentation models, such as brain tumor or stroke lesion segmentation. Hopefully, the work presented in this thesis will open the door and motivate further deep learning research for the task of MS lesion detection in particular, and brain imaging in general.

References

[1] *Deep learning tutorials, http://deeplearning.net/reading-list/tutorials/.*

[2] *Government of canada, statistics canada, table105-13001,neurological conditions, by age group and sex, household population aged 0 and over, 2010/2011.*

[3] *Multiple sclerosis foundation, http://www.msfocus.org/.*

[4] *Multiple sclerosis information on healthline, medical information for healthy living http://www.healthline.com/adamcontent/multiple-sclerosis/.*

[5] *Multiple sclerosis society of canada, https://mssociety.ca.*

[6] *Nhcc, mapping connections : an understanding of neurological conditions in canada.*

[7] Ayelet Akselrod-Ballin, Meirav Galun, Ronen Basri, Achi Brandt, Moshe John Gomori, Massimo Filippi, and Paula Valsasina, *An integrated segmentation and classification approach applied to multiple sclerosis analysis*, Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, vol. 1, IEEE, 2006, pp. 1122–1129.

[8] Ayelet Akselrod-Ballin, Meirav Galun, John Moshe Gomori, Massimo Filippi, Paula Valsasina, Ronen Basri, and Achi Brandt, *Automatic segmentation and classification of multiple sclerosis in multichannel mri*, Biomedical Engineering, IEEE Transactions on **56** (2009), no. 10, 2461–2469.

[9] Petronella Anbeek, Koen L Vincken, Matthias JP van Osch, Robertus HC Bisschops, and Jeroen van der Grond, *Probabilistic segmentation of white matter lesions in mr imaging*, NeuroImage **21** (2004), no. 3, 1037–1044.

[10] Petronella Anbeek, Koen L Vincken, and Max A Viergever, *Automated ms-lesion segmentation by k-nearest neighbor classification*, The MIDAS Journal-MS Lesion Segmentation (MICCAI 2008 Workshop), 2008.

[11] Rohit Bakshi, Suzie Ariyaratana, Ralph HB Benedict, and Lawrence Jacobs, *Fluid-attenuated inversion recovery magnetic resonance imaging detects cortical and juxtacortical multiple sclerosis lesions*, Archives of neurology **58** (2001), no. 5, 742–748.

[12] P-L Bazin and Dzung L Pham, *Topology-preserving tissue classification of magnetic resonance brain images*, Medical Imaging, IEEE Transactions on **26** (2007), no. 4, 487–496.

[13] Yoshua Bengio, Ian J. Goodfellow, and Aaron Courville, *Deep learning*, Book in preparation for MIT Press, 2015.

[14] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle, *Greedy layer-wise training of deep networks*.

[15] Tom Brosch, Roger Tam, Alzheimers Disease Neuroimaging Initiative, et al., *Manifold learning of brain mris by deep learning*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013, Springer, 2013, pp. 633–640.

[16] Antonio Cerasa, Eleonora Bilotta, Antonio Augimeri, Andrea Cherubini, Pietro Pantano, Giancarlo Zito, Pierluigi Lanza, Paola Valentino, Maria C Gioia, and Aldo Quattrone, *A cellular neural network methodology for the automated segmentation of multiple sclerosis lesions*, Journal of neuroscience methods **203** (2012), no. 1, 193–199.

[17] Dan Ciresan, Alessandro Giusti, Luca M Gambardella, and Jürgen Schmidhuber, *Deep neural networks segment neuronal membranes in electron microscopy images*, Advances in neural information processing systems, 2012, pp. 2843–2851.

[18] Chris A. Cocosco, Vasken Kollokian, Remi K.-S. Kwan, G. Bruce Pike, and Alan C. Evans, *Brainweb: Online interface to a 3d mri simulated brain database*, NeuroImage **5** (1997), 425.

[19] D Louis Collins, Colin J Holmes, Terrence M Peters, and Alan C Evans, *Automatic 3-d model-based neuroanatomical segmentation*, Human brain mapping **3** (1995), no. 3, 190–208.

[20] D Louis Collins, Peter Neelin, Terrence M Peters, and Alan C Evans, *Automatic 3d intersubject registration of mr volumetric data in standardized talairach space.*, Journal of computer assisted tomography **18** (1994), no. 2, 192–205.

[21] D Louis Collins, Alex P Zijdenbos, Wim FC Baaré, and Alan C Evans, *Animal+ insect: improved cortical structure segmentation*, Information Processing in Medical Imaging, Springer, 1999, pp. 210–223.

[22] Angel Alfonso Cruz-Roa, John Edison Arevalo Ovalle, Anant Madabhushi, and Fabio Augusto González Osorio, *A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013, Springer, 2013, pp. 403–410.

[23] Li Deng and Dong Yu, *Deep learning: methods and applications*, Foundations and Trends in Signal Processing **7** (2014), no. 3–4, 197–387.

[24] Richard O Duda, Peter E Hart, et al., *Pattern classification and scene analysis*, vol. 3, Wiley New York, 1973.

[25] Colm Elliott, Douglas L Arnold, D Louis Collins, and Tal Arbel, *Temporally consistent probabilistic detection of new multiple sclerosis lesions in brain mri*, Medical Imaging, IEEE Transactions on **32** (2013), no. 8, 1490–1503.

[26] F Fazekas, F Barkhof, M Filippi, RI Grossman, DKB Li, WI McDonald, HF McFarland, DW Paty, JH Simon, JS Wolinsky, et al., *The contribution of magnetic resonance imaging to the diagnosis of multiple sclerosis*, Neurology **53** (1999), no. 3, 448–448.

[27] S.J. Francis, *Automatic lesion identification in mri of multiple sclerosis patients*, Master's thesis, McGill University, 2004.

[28] Robin JM Franklin et al., *Remyelination in the cns: from biology to therapy*, Nature Reviews Neuroscience **9** (2008), no. 11, 839–855.

[29] Daniel García-Lorenzo, Simon Francis, Sridar Narayanan, Douglas L Arnold, and D Louis Collins, *Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging*, Medical image analysis **17** (2013), no. 1, 1–18.

[30] Daniel García-Lorenzo, Sylvain Prima, Douglas L Arnold, D Louis Collins, and Christian Barillot, *Trimmed-likelihood estimation for focal lesions and tissue segmentation in multisequence mri for multiple sclerosis*, Medical Imaging, IEEE Transactions on **30** (2011), no. 8, 1455–1467.

[31] Jun Ge, Berkman Sahiner, Lubomir M Hadjiiski, Heang-Ping Chan, Jun Wei, Mark A Helvie, and Chuan Zhou, *Computer aided detection of clusters of microcalcifications on full field digital mammograms*, Medical physics **33** (2006), no. 8, 2975–2988.

[32] Ezequiel Geremia, Bjoern H Menze, Olivier Clatz, Ender Konukoglu, Antonio Criminisi, and Nicholas Ayache, *Spatial decision forests for ms lesion segmentation in multi-channel mr images*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010, Springer, 2010, pp. 111–118.

[33] D Goldberg-Zimring, A Achiron, S Miron, M Faibel, and H Azhari, *Automated detection and characterization of multiple sclerosis lesions in brain mr images*, Magnetic resonance imaging **16** (1998), no. 3, 311–318.

[34] Mehdi Habibzadeh, Adam Krzyżak, and Thomas Fevens, *White blood cell differential counts using convolutional neural networks for low resolution images*, Artificial Intelligence and Soft Computing, Springer, 2013, pp. 263–274.

[35] Andreas Hadjiprocopis and Paul Tofts, *An automatic lesion segmentation method for fast spin echo magnetic resonance images using an ensemble of neural networks*, Neural Networks for Signal Processing, 2003. NNSP'03. 2003 IEEE 13th Workshop on, IEEE, 2003, pp. 709–718.

[36] Rola Harmouche, Louis Collins, Douglas Arnold, Simon Francis, and Tal Arbel, *Bayesian ms lesion classification modeling regional and local spatial information*, Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, vol. 3, IEEE, 2006, pp. 984–987.

[37] Rola Harmouche, Nagesh K Subbanna, D Louis Collins, Douglas L Arnold, and Tal Arbel, *Probabilistic multiple sclerosis lesion classification based on modeling regional intensity variability and local neighborhood information*, Biomedical Engineering, IEEE Transactions on **62** (2015), no. 5, 1281–1292.

[38] Kaiming He and Xiangyu Zhang, *Delving deep into rectifiers: Surpassing human-level performance on imagenet classification*, arXiv preprint arXiv:1502.01852 (2015).

[39] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, *Spatial pyramid pooling in deep convolutional networks for visual recognition*, arXiv preprint arXiv:1406.4729 (2014).

[40] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, *A fast learning algorithm for deep belief nets*, Neural computation **18** (2006), no. 7, 1527–1554.

[41] Kurt Hornik, Maxwell Stinchcombe, and Halbert White, *Multilayer feedforward networks are universal approximators*, Neural networks **2** (1989), no. 5, 359–366.

[42] David H Hubel and Torsten N Wiesel, *Receptive fields, binocular interaction and functional architecture in the cat's visual cortex*, The Journal of physiology **160** (1962), no. 1, 106–154.

[43] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, *3d convolutional neural networks for human action recognition*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **35** (2013), no. 1, 221–231.

[44] B Johnston, M Stella Atkins, B Mackiewich, and M Anderson, *Segmentation of multiple sclerosis lesions in intensity corrected multispectral mri*, Medical Imaging, IEEE Transactions on **15** (1996), no. 2, 154–169.

[45] Micheline Kamber, Rajjan Shinghal, D Louis Collins, Gordon S Francis, and Alan C Evans, *Model-based 3-d segmentation of multiple sclerosis lesions in magnetic resonance brain images*, Medical Imaging, IEEE Transactions on **14** (1995), no. 3, 442–453.

[46] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, *Large-scale video classification with convolutional neural networks*, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1725–1732.

[47] Minjeong Kim, Guorong Wu, and Dinggang Shen, *Unsupervised deep learning for hippocampus segmentation in 7.0 tesla mr images*, Machine Learning in Medical Imaging, Springer, 2013, pp. 1–8.

[48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, *Imagenet classification with deep convolutional neural networks*, Advances in neural information processing systems, 2012, pp. 1097–1105.

[49] Dirk-Jan Kroon, van ESB Oort, and CH Slump, *Multiple sclerosis detection in multispectral magnetic resonance images with principal components analysis*, (2008).

[50] Hugo Larochelle, *Neural networks class*, 2014.

[51] Quoc V Le, Will Y Zou, Serena Y Yeung, and Andrew Y Ng, *Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis*, Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE, 2011, pp. 3361–3368.

[52] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel, *Backpropagation applied to handwritten zip code recognition*, Neural computation **1** (1989), no. 4, 541–551.

[53] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE **86** (1998), no. 11, 2278–2324.

[54] Yann LeCun, Corinna Cortes, and Christopher JC Burges, *The mnist database of handwritten digits*, 1998.

[55] Rongjian Li, Wenlu Zhang, Heung-Il Suk, Li Wang, Jiang Li, Dinggang Shen, and Shuiwang Ji, *Deep learning based imaging data completion for improved brain disease diagnosis*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014, Springer, 2014, pp. 305–312.

[56] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang, *Deepreid: Deep filter pairing neural network for person re-identification*, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 152–159.

[57] Shu Liao, Yaozong Gao, Aytekin Oto, and Dinggang Shen, *Representation learning: a unified deep learning framework for automatic prostate mr segmentation*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013, Springer, 2013, pp. 254–261.

[58] Siqi Liu, Sidong Liu, Weidong Cai, Sonia Pujol, Ron Kikinis, and Dagan Feng, *Early diagnosis of alzheimer's disease with deep learning*, Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on, IEEE, 2014, pp. 1015–1018.

[59] JB Antoine Maintz and Max A Viergever, *A survey of medical image registration*, Medical image analysis **2** (1998), no. 1, 1–36.

[60] J Morra, Zhuowen Tu, A Toga, and P Thompson, *Automatic segmentation of ms lesions using a contextual model for the miccai grand challenge*, Grand Challenge Work.: Mult. Scler. Lesion Segm. Challenge (2008), 1–7.

[61] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng, *Reading digits in natural images with unsupervised feature learning*, NIPS workshop on deep learning and unsupervised feature learning, vol. 2011, Granada, Spain, 2011, p. 5.

[62] László G Nyúl, Jayaram K Udupa, and Xuan Zhang, *New variants of a method of mri scale standardization*, Medical Imaging, IEEE Transactions on **19** (2000), no. 2, 143–150.

[63] Sinno Jialin Pan and Qiang Yang, *A survey on transfer learning*, Knowledge and Data Engineering, IEEE Transactions on **22** (2010), no. 10, 1345–1359.

[64] Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio, *How to construct deep recurrent neural networks*, arXiv preprint arXiv:1312.6026 (2013).

[65] Boris Teodorovich Polyak, *Some methods of speeding up the convergence of iteration methods*, USSR Computational Mathematics and Mathematical Physics **4** (1964), no. 5, 1–17.

[66] Adhish Prasoon, Kersten Petersen, Christian Igel, François Lauze, Erik Dam, and Mads Nielsen, *Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013, Springer, 2013, pp. 246–253.

[67] Holger R Roth, Le Lu, Ari Seff, Kevin M Cherry, Joanne Hoffman, Shijun Wang, Jiamin Liu, Evrim Turkbey, and Ronald M Summers, *A new 2.5 d representation for lymph node detection using random sets of deep convolutional neural network observations*, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014, Springer, 2014, pp. 520–527.

[68] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, *ImageNet Large Scale Visual Recognition Challenge*, International Journal of Computer Vision (IJCV) (2015), 1–42.

[69] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., *Imagenet large scale visual recognition challenge*, arXiv preprint arXiv:1409.0575 (2014).

[70] R. Muthuganapathy G. Krishnamurthi S. Vaidya, A. Chunduru, *Longitudinal multiple sclerosis lesion segmentation using 3d convolutional neural networks*, The ISBI Longitudinal MS Lesion Segmentation Challenge (2015).

[71] Ruslan Salakhutdinov and Geoffrey E Hinton, *Deep boltzmann machines*, International Conference on Artificial Intelligence and Statistics, 2009, pp. 448–455.

[72] M. Scully, V. Magnotta, C. Gasparovic, P. Pelligrimo, D. Feis, and H. Bockholt, *3d segmentation in the clinic: A grand challenge ii at miccai 2008 - ms lesion segmentation*, (2008).

[73] Pierre Sermanet, Soumith Chintala, and Yann LeCun, *Convolutional neural networks applied to house numbers digit classification*, Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE, 2012, pp. 3288–3291.

[74] Navid Shiee, Pierre-Louis Bazin, Arzu Ozturk, Daniel S Reich, Peter A Calabresi, and Dzung L Pham, *A topology-preserving approach to the segmentation of brain images with multiple sclerosis lesions*, NeuroImage **49** (2010), no. 2, 1524–1535.

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

[76] John G Sled, Alex P Zijdenbos, and Alan C Evans, *A nonparametric method for automatic correction of intensity nonuniformity in mri data*, Medical Imaging, IEEE Transactions on **17** (1998), no. 1, 87–97.

[77] Stephen M Smith, *Fast robust automated brain extraction*, Human brain mapping **17** (2002), no. 3, 143–155.

[78] Jean-Christophe Souplet, Christine Lebrun, Nicholas Ayache, and Grégoire Malandain, *An automatic segmentation of t2-flair multiple sclerosis lesions*, The MIDAS Journal-MS Lesion Segmentation (MICCAI 2008 Workshop), 2008.

[79] Martin Styner, Joohwi Lee, Brian Chin, M Chin, Olivier Commowick, H Tran, S Markovic-Plese, V Jewells, and S Warfield, *3d segmentation in the clinic: A grand challenge ii: Ms lesion segmentation*, MIDAS Journal **2008** (2008), 1–6.

[80] N Subbanna, M Shah, SJ Francis, S Narayanan, DL Collins, DL Arnold, and T Arbel, *Ms lesion segmentation using markov random fields*, Proceedings of International Conference on Medical Image Computing and Computer Assisted Intervention, London, UK, 2009.

[81] Nagesh Subbanna, Doina Precup, Douglas Arnold, and Tal Arbel, *Image: Iterative multilevel probabilistic graphical model for detection and segmentation of multiple sclerosis lesions in brain mri*, Information Processing in Medical Imaging, Springer, 2015, pp. 514–526.

[82] Yi Sun, Xiaogang Wang, and Xiaoou Tang, *Deep convolutional network cascade for facial point detection*, Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE, 2013, pp. 3476–3483.

[83] Tang Xiaoou Sun Yi, Wang Xiaogang, *Deep learning face representation from predicting 10,000 classes*, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1891–1898.

[84] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton, *On the importance of initialization and momentum in deep learning*, Proceedings of the 30th International Conference on Machine Learning (ICML-13), 2013, pp. 1139–1147.

[85] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, *Going deeper with convolutions*, arXiv preprint arXiv:1409.4842 (2014).

[86] Alexander Toshev and Christian Szegedy, *Deeppose: Human pose estimation via deep neural networks*, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1653–1660.

[87] Koen Van Leemput, Frederik Maes, Dirk Vandermeulen, Alan Colchester, and Paul Suetens, *Automated segmentation of multiple sclerosis lesions by model outlier detection*, Medical Imaging, IEEE Transactions on **20** (2001), no. 8, 677–688.

[88] Michael W Vannier, RL Butterfield, DL Rickman, DM Jordan, WA Murphy, and PR Biondetti, *Multispectral magnetic resonance image analysis.*, Critical reviews in biomedical engineering **15** (1986), no. 2, 117–144.

[89] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, *Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion*, The Journal of Machine Learning Research **11** (2010), 3371–3408.

[90] Simon Vinitski, Carlos F Gonzalez, Robert Knobler, David Andrews, Tad Iwanaga, and Mark Curtis, *Fast tissue segmentation based on a 4d feature map in characterization of intracranial lesions*, Journal of Magnetic Resonance Imaging **9** (1999), no. 6, 768–776.

[91] Li Wan, Matthew Zeiler, Sixin Zhang, Yann L Cun, and Rob Fergus, *Regularization of neural networks using dropconnect*, Proceedings of the 30th International Conference on Machine Learning (ICML-13), 2013, pp. 1058–1066.

[92] Simon K Warfield, Michael Kaus, Ferenc A Jolesz, and Ron Kikinis, *Adaptive, template moderated, spatially varying statistical classification*, Medical image analysis **4** (2000), no. 1, 43–55.

[93] Michael Wels, Martin Huber, and Joachim Hornegger, *Fully automated segmentation of multiple sclerosis lesions in multispectral mri*, Pattern Recognition and Image Analysis **18** (2008), no. 2, 347–350.

[94] Ying Wu, Simon K Warfield, I Leng Tan, William M Wells, Dominik S Meier, Ronald A van Schijndel, Frederik Barkhof, and Charles RG Guttmann, *Automated segmentation of multiple sclerosis lesion subtypes with multichannel mri*, NeuroImage **32** (2006), no. 3, 1205–1215.

[95] Yan Xu, Tao Mo, Qiwei Feng, Peilin Zhong, Maode Lai, Eric I Chang, et al., *Deep learning of feature representation with multiple instance learning for medical image analysis*, Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on, IEEE, 2014, pp. 1626–1630.

[96] Daisuke Yamamoto, Hidetaka Arimura, Shingo Kakeda, Taiki Magome, Yasuo Yamashita, Fukai Toyofuku, Masafumi Ohki, Yoshiharu Higashida, and Yukunori Korogi, *Computer-aided detection of multiple sclerosis lesions in brain magnetic resonance images: False positive reduction scheme consisted of rule-based, level set method, and support vector machine*, Computerized Medical Imaging and Graphics **34** (2010), no. 5, 404–413.

[97] Akmal A Younis, Ahmed T Soliman, Mansur R Kabuka, and Nigel M John, *Ms lesions detection in mri using grouping artificial immune networks*, Bioinformatics and Bioengineering, 2007. BIBE 2007. Proceedings of the 7th IEEE International Conference on, IEEE, 2007, pp. 1139–1146.

[98] Alex P Zijdenbos, Benoit M Dawant, Richard A Margolin, and Andrew C Palmer, *Morphometric analysis of white matter lesions in mr images: method and validation*, Medical Imaging, IEEE Transactions on **13** (1994), no. 4, 716–724.

[99] Alex P Zijdenbos, Reza Forghani, and Alan C Evans, *Automatic pipeline analysis of 3-d mri data for clinical trials: application to multiple sclerosis*, Medical Imaging, IEEE Transactions on **21** (2002), no. 10, 1280–1291.