

Detection of Multiple Sclerosis Using Deep Learning

by

Sabila Al Jannat
17101302

Tanjina Hoque
17101129

Nafisa Alam Supti
16201083

A thesis submitted to the Department of Computer Science and Engineering in
partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
January 2021

© 2021. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sabila Al Jannat

Sabila Al Jannat
17101302

Tanjina Hoque

Tanjina Hoque
17101129

Nafisa Alam Supti

Nafisa Alam Supti
16201083

Approval

The thesis/project titled “Detection of Multiple Sclerosis using Deep Learning” submitted by

- 1.Sabila Al Jannat (17101302)
- 2.Tanjina Hoque (17101129) 3.Nafisa Alam Supti (16201083)

Of Fall, 2020 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on January 8, 2021.

Examining Committee:

Supervisor:



Md. Ashraful Alam, PhD
Assistant Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:



Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:

Mahbubul Alam Majumdar, PhD Professor
and Dean, School of Data and Sciences Department of
Computer Science and Engineering
Brac University

Abstract

Accurate detection of white matter lesions in 3D Magnetic Resonance Images (MRIs) of patients with Multiple Sclerosis is essential for diagnosis and treatment evaluation of MS. It is strenuous for the optimal treatment of the disease to detect early MS and estimate its progression. In this study, we propose efficient Multiple Sclerosis detection techniques to improve the performance of a supervised machine learning algorithm and classify the progression of the disease. Detection of MS lesions become more intricate due to the presence of unbalanced data with a very small number of lesions pixel. Our pipeline is evaluated on MS patients data from the Laboratory of Imaging Technologies. Fluid-attenuated inversion recovery (FLAIR) series are incorporated to introduce a faster system alongside maintaining readability and accuracy. Our approach is based on convolutional neural networks (CNN). We have trained the model using transfer learning and used softmax as an activation function to classify the progression of the disease. Our results significantly show the effectiveness of the usage of MRI of MS lesions. Experiments on 30 patients and 100 healthy brain MRIs can accurately predict disease progression. Manual detection of lesions by clinical experts is complicated and time-consuming as a large amount of MRI data is required to analyze. We analyze the accuracy of the proposed model on the dataset. Our approach exhibits a significant accuracy rate of up to 98.24%.

Keywords: Magnetic Resonance Imaging(MRI); Machine Learning; Multiple Sclerosis(MS); 3D Magnetic Resonance Imaging; White Matter Lesion Detection; Deep Learning; Convolutional Neural Network(CNN); Fluid-attenuated inversion recovery (FLAIR); Data Augmentation; Image Processing

Acknowledgement

We would like to express our sincere gratitude to The Most Merciful Allah who blessed us with patience, determination, ability and opportunity to complete the research work within time.

We are highly indebted for getting such a tremendous opportunity to prepare the report on Detecting Multiple Sclerosis using Deep Learning. We would like to thank wholeheartedly our supervisor, Md. Ashraf Alam, PhD (Assistant Professor), Department of Computer Science and Engineering, BRAC University for giving us guidelines about how we can proceed prepare this report.

Finally, we are grateful to all of our family members and faculty members of Department of Computer Science and Engineering, BRAC University for believing in our ability and for supporting and encouraging us through thick and thin.

Table of Contents

Declarationi Approvalii

Abstractiii

Acknowledgmentiv

Table of Contentsv List

of Figuresvii List of

Tablesviii

Nomenclatureix

1 Introduction1

1.1 Motivation.....	1
1.2 Problem Statement.....	1
1.3 Aims and Objectives.....	2
1.4 Thesis Outline.....	2

2 Related Works3

3 Materials and Methodology6

3.1 Data Acquisition and Preparation.....	6
3.2 Image Pre-processing.....	7
3.3 Data Augmentation.....	7
3.4 Data Splitting.....	9

4 Prediction Modeling using CNN10

4.1 Convolutional Neural Network.....	10
4.2 Transfer Learning.....	12
4.3 Classification.....	13
4.4 Model Compilation.....	13

5 Result and Analysis16

5.1 Comparison.....	16
5.2 Performance Efficiency.....	17

6 Conclusion and Future Work20

Bibliography23

List of Figures

3.1	Image Resizing	7
3.2	Color channel swap (BGR to RGB).....	8
3.3	Data Augmentation using rotation technique.....	8
3.4	Work Flow	9
4.1	Implementation of Convolution layers	11
4.2	Diagram of Convolutional Neural Network.....	11
4.3	Before Transfer Learning	12
4.4	After Transfer Learning	13
4.5	Diagram of fully-connected layer	14
4.6	Summary of the model.....	15
5.1	Detection Results from VGG16 model.....	17
5.2	Normalized Confusion Matrix	18
5.3	Training loss and accuracy curve	19

List of Tables

3.1	Demographic Characteristics of Dataset.....	6
3.2	Number of Slices in Datasets.....	9
4.1	Size of Input, Filter and Output	10
5.1	The obtained results from different top rank strategies	16
5.2	The obtained results from different strategies of CNN.....	16

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AM – FM Amplitude- Modulation, Frequency-Modulation

CIS Clinically Isolated Symptom

CNN Convolutional Neural Network *FLAIR*

Fluid-attenuated inversion recovery *GLCM*

Grey Level Co-occurrence Matrix *MRI*

Magnetic Resonance Imaging

MS Multiple Sclerosis

PR Progressive-relapsing

RR Relapsing-remitting

SP Secondary Progressive

SVM Support Vector Machine

Chapter 1

Introduction

1.1 Motivation

Magnetic Resonance Imaging (MRI) synthesis has gained attention due to its diverse applications in medical diagnostics. And so, in the detection of the white matter MS lesions of the brain or spinal cord, MRIs are considered to be examined. Researchers have found that MS is one of the most common primary neurological disorders in young adults[12]. The detection of MS is of major importance in medical diagnostics at present. As of yet, there is little to no cure for MS. However, early treatments can improve recovery from attacks and control the symptoms. Detecting white matter lesions using deep learning from brain MRIs may lead to improvement of understanding the progression of the disease. Deep learning is considered the most assured and well-known ML technique for radiology in particular. Being a subset of Machine Learning, Deep Learning can strengthen imaging diagnosis processes.

1.2 Problem Statement

Multiple Sclerosis is a chronic immune disease of the central nervous system which appears in areas of the brain and spinal cord with the occurrence of focal regions of inflammatory demyelination[2], [27]. Myelin sheath plays a significant role in signal transmission between neurons[27]. Disruption of the structure of the myelin sheath affects the functioning of the organ[27]. It can affect the visual system and create hindrance in mobility. There is no definite treatment for MS but early diagnosis and treatment can decrease the progression of the disease. After following different ways of detection like the evoked potential test, blood testing, lumbar puncture, etc. manual methods of detecting MS lesions from MRI results in a decision. MRIs to detect the lesions are used to diagnose the disease and track its progress by clinicians and radiologists, which is a very time-consuming process and suffers from variability. Detection of MS lesions from MRI may be misleading as it is a sensitive method itself. So it has become a necessity to automatize the process. Deep learning provides a significant contribution to medical diagnosis. We aimed to evaluate the diagnostic accuracy using a deep learning algorithm in detecting multiple sclerosis. Medical image processing through AI may bring a better solution by addressing this problem.

1.3 Aims and Objectives

We aim to introduce an automatic intelligent system of MS lesion detection. Because manual analysis is time-consuming, inefficient, and expensive. Deep Learning and image processing techniques have the capability of medical research for clinical studies on a large-scale. The objective of our approach is to develop a neural network-based decision system that detects white matter MS lesions with a high degree of accuracy. Therefore, we introduce an efficient technique to detect MS lesions from MRIs of MS patients.

1.4 Thesis Outline

Chapter One introduces the treatise of Deep Learning, the necessity of detection of MS using deep learning, the aim and objectives of this paper, and the motivation behind the work.

Chapter Two discusses the related works on the field with MRIs detecting white matter MS lesions of MS patients.

Chapter Three mentions the data collection sources we have used and the materials and methodology of the work.

Chapter Four discusses the prediction model CNN to detect white matter MS lesions from MRI scans.

Chapter Five provides a comparison between the accuracy given by different approaches and the analysis of the result.

Chapter Six concludes the work and discusses the inspiration for future work on the field.

Chapter 2

Related Works

There has been a variety of approaches to detect MS lesions from MRIs in the literature. In the past few years, researchers have developed numerous algorithms in MS detection.

The approach proposed by Liu et. al.[4] is an automatic process that detects Multiple Sclerosis from multi-modal MRIs like T2-weighted and FLAIR images. The major point of the approach is to treat the Multiple Sclerosis lesions as outliers to the normal brain tissue distribution by optimizing L2E measure. Here, L2E measure is the square of the difference between the true density and the assumed Gaussian mixture. Leemput et. al.[1] also treats MS lesions as an outlier class. But in this paper, to treat MS lesions as outliers, a stochastic model performs intensity-based tissue classification for normal brain images. The outcome of the approach is further compared with the lesion delineations performed by human experts which shows a high total lesion load correlation.

In[15], Samah et. al. focused on MS lesions detection from noisy MRIs tissue. This paper introduced a new method named Decimal Descriptor Patterns (DDP) for MRI analysis. The classification here is done based on the classifier SVM. In this paper, two approaches of texture analysis named the Grey Level Co-occurrence Matrix (GLCM) and the Local Binary Patterns (LBP) are used as references for comparison in image analysis. A similar procedure was adopted by Zhou et. al.[17]. In this study, GLCM is used for feature extraction and Multi-layered feedforward neural network is used as the classifier. A biogeography-based optimization algorithm was used to train this classifier. Washimkar et. al.[11] also worked on texture analysis to find the progression of the disease. But in this paper, the authors followed the segmentation and feature extraction approach. Then further by utilizing the Saliency map method, AM-FM segmentation is completed. Moreover, by utilizing the Fuzzy C means clustering method, the filtered segmented features are clustered. Finally, the feature extraction techniques are used to detect the features and classify them by using the K-NN classifier.

Many image processor techniques are adopted and used to detect the MS lesions[7].

The adopted methods are to segment brain tissue from the skull bones. To producing thin edges and connected boundaries, the Marr-Hildreth method is adopted. A two-input two-output fully CNN model for MS lesion synthesis in MRI was proposed to detect MS lesions[23]. This technique improved the performance of the SVM algorithm.

An unsupervised approach can also detect MS lesions from MRI scans. In[6], Zeng

et. al. proposed an unsupervised approach for MS lesions segmentation in multi-modal MRIs. The proposed approach is based on joint histogram modeling followed by false-positive reduction and alpha matting. Another unique approach named Cascaded CNN has been proposed in[1] for WM MS lesion segmentation of MS patients MRI. Cascaded CNN is an approach based on a cascade of two 3D patch-wise CNN and is very well known for its capability to learn from a small training dataset. Jain et. al. in the paper[5] introduced a technique, MSmetrix. MSmetrix is an accurate and automatic method for MS lesion segmentation from MRI scans. This technique is independent of scanner or acquisition protocol. MSmetrix can be implemented without any training data.

Brosch et. al.[8] proposed a 3D convolutional encoder networks approach with shortcut connections for MS lesions segmentation from MRI scans. The model used in this paper is a neural network consisting of two interconnected pathways: Convolutional pathways and Deconvolutional Pathways. Convolutional pathway learns progressively more theoretical and high-end image features. But on the other hand, the Deconvolutional pathway predicts the final segmentation at the voxel level.

La Rosa et. al.[25] proposed a fully-convolutional deep learning approach based on the 3D U-Net. This method worked in segmenting cortical and white matter lesions at 3T automatically. In[13] Aslani et. al. presented an automated approach for MS lesions segmentation from multi-modal brain MRI Scans. The method used in this paper is based on a deep end-to-end 2D CNN to segment 3D volumetric data based on slices. The proposed model develops the network to encoding information from multiple modalities individually using a multi-branch down-sampling path.

In the paper[20], Kats et. al. proposed the use of a soft ground-truth mask which is also known as a soft mask for training Fully Convolutional Neural Network to segment MS lesions. This paper focused on using the anatomical knowledge present in the pixels that surround the lesions by increasing the data set of the lesion class with neighboring pixel data and decreasing the confidence weight as well. They have also shown that softmax can improve the performance of network segmentation while comparing it to a second independent expert.

Akselrod-Ballin et. al.[3] established an approach that is a combination of both segmentation and classification. The paper aims to detect abnormal brain tissues in medical brain images. The approach results in the detection of MS lesions from 3D multichannel MRIs. The segmentation process of the approach produces a rich set of features like location, intensity, neighborhood relations, shape, and anatomical context. These features are further passed through the decision forest classifier which is trained with data labels. This approach uses regional properties for characterizing the brain tissues more efficiently.

Zhang et. al.[16] in this paper developed an improved CNN technique. The improved CNN approach is a combination of the parametric rectified linear unit also known as PReLU and dropout techniques. 10 layer deep convolutional neural network along with 7 convolutional layers and 3 fully connected layers have been used in the process. PReLU and the dropout technique helped to increase the accuracy which results in a superior method to four state-of-art approaches. In the paper[21], Narayana et. al. also proposed a technique using CNN for classification to enhance the lesions. And the classification was performed on each slice. The network performance in this paper was evaluated using five-fold cross-validation.

In the paper[28], Sepahvand et. al. introduced an automatic deep learning technique

that aims to detect and segment NE T2w lesions from longitudinal brain MRIs of RR MS patients. The technique is adapted from 3D U-net and is also based on subtraction MRI that helps to individualize between real anatomical change and artifactual change while searching for small lesions. Moreover, the approach also classifies the patients on the basis of their activity or inactivity.

Doğan et. al.[19] presents machine learning techniques like k means and support vector machine. K-means is an unsupervised machine learning technique whereas support vector machine is a supervised machine learning technique. In this paper, the dataset used had no labels so the labels were generated manually by pixel values from the original MRIs. In the future Doğan et. al.[19] aims to automate the segmentation method.

In our paper, we have proposed a CNN technique where we trained the model using transfer learning technique and used the softmax function to classify the MS lesions from MRI images according to the progression of MS disease.

Chapter 3

Materials and Methodology

3.1 Data Acquisition and Preparation

The dataset of the Laboratory of Imaging Technologies[30] consists of 3D NIfTI images of 30 MS patients. Each case has 4 modalities: T1-weighted, T2-weighted, T1-wks, and FLAIR. The dataset consists of a total of 3766 slices of patient brain MR images from 30 MS patients. We have also used 100 slices of the healthy brain MRIs[18]. In our approach, we work with a subset of MRI scans from the original dataset. In this dataset[30], there are samples of MRIs having MS lesions except for one. The dimension of each volume is 512 192 3. Images from each sample were used as a separate sample for experimentation. The dataset was split such that all the scans from each sample were placed into one of two, non-overlapping groups: training and test sets. There are 5 types of a label in the dataset: CIS(Clinically Iso- lated Syndrome), RR MS(Relapsing-remitting MS), PR MS(Progressive-relapsing MS), SP MS(Secondary Progressive MS) and NO(No MS). There are all the cate- gories of MS patients. In the dataset, a maximum number of patients suffering are from the RR MS category. Here, a patient falls in the CIS category, when a patient faces the first episode of symptoms[29]. RR MS patients have attacks of symptoms from time to time, followed by weeks, months, or years of recovery(remission)[29]. SP MS patients have no well-defined attacks or symptoms, and there is almost no remissions[29]. The subjects were between 25 and 64 years of age. The dataset contains 30 patients where 24 were RR MS patients, 2 were SP MS patients, 2 CIS patients, 1 were PR MS patients, 1 was unidentified. And there were 100 healthy brains MRI in a different dataset.

Table 3.1: Demographic Characteristics of Dataset

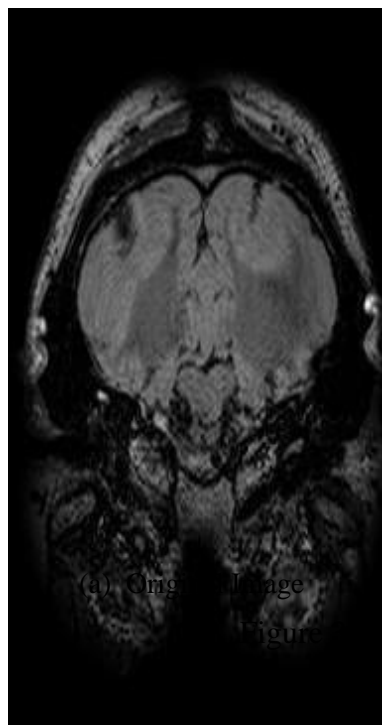
No. of Subjects	Age	Gender(F/M)	No. of Slices
30(MS)	44.5±19.5	23/7	3766
100(Healthy)	–	–	100

The MRI scan can produce a 3D model of the body. It is difficult to perform image processing in 3D MRIs[10]. Therefore, it is required to transform the series of 3D MRI images into a series of 2D images before image pre-processing[10]. We worked with 2D images that were converted from 3D MRI scans. Only the FLAIR MRI scans were considered for this study. Series of 2D images were processed before image

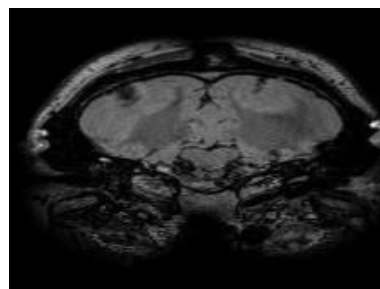
pre-processing. We have performed the processing steps using OpenCV library[10]. Then we split the data into the multi-class folder(RR, PR, SP, CIS, NO). We also represented multi-class labels of the brain MRI scans to binary labels by using LabelBinarizer.

3.2 Image Pre-processing

To achieve the best result processed images were further processed. After we have loaded the image we have swapped color channels from BGR to RGB (Figure 3.2). RGB channel is composed of three colors: red, green, and blue. Each RGB image stores discrete pixels with brightness intensities between 0 and 255. All the images which were considered as an input to the CNN model were resized into 224 224 dimension (Figure 3.1). Because the presence of different sizes in the dataset may reduce the accuracy of the classification. Initially, we have gathered all the test images in a list of images for testing. We have also extracted the class label from the filename using the os library and made a list of labels out of it. We calculate the means of labels and apply those means by replacing missing values with the means.



(a) Original Image

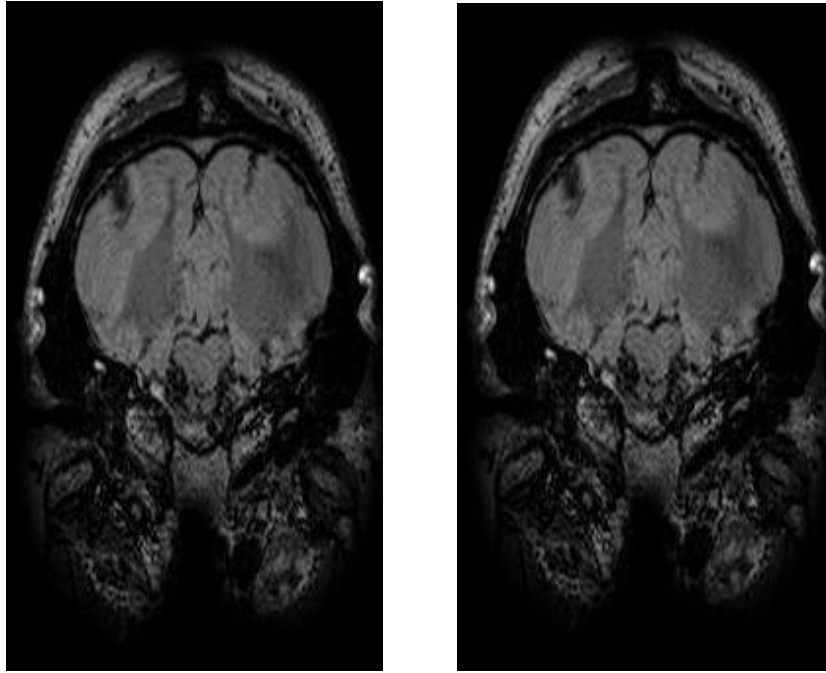


(b) Resized Image

Figure 3.1 Image Resizing

3.3 Data Augmentation

The collection of good data is very challenging in some cases. Insufficient good data can slow down and hinder the development of a model. Data Augmentation is one of the ways to get around the insufficiency of good data. Data Augmentation facilitates the development of the model by preparing far more data than we have.



(a) BGR

(b) RGB

Figure 3.2: Color channel swap (BGR to RGB)

Image Augmentation is performed as a part of the pre-processing step. As it is usually preferred for small datasets. One common Data Augmentation approach was implemented on the training dataset. The Data Augmentation approach was image rotation. The rotation angle θ was set to 15° (Figure 3.3).

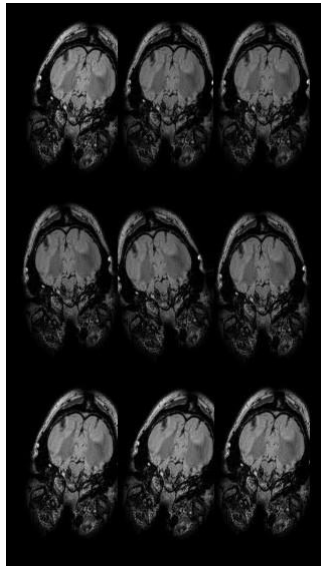


Figure 3.3: Data Augmentation using rotation technique

Due to this approach, we could generate 9 new images for each original image in the training set. As a result, the size of the training set was enhanced by 9-fold.

3.4 Data Splitting

The dataset was split in such a way that all the MRI scans from each subject were placed into one of two groups: training and testing sets. In the training set, we have 3013 images of MS patients and 80 images of healthy persons. In the test set, we have 753 images of MS patients and 20 images of healthy persons. The dataset is divided randomly into 80% for training and 20% for testing. And we set the stratify parameter to labels so that the train test split method returns training and testing subsets in such a manner that the same proportion of class labels are provided as the input dataset. In the train test split method, we have also set a fixed value to the random state parameter i.e. 42. Because no matter how many times we execute the code the result is expected to be the same as the same values of train and test datasets will be considered. And we have changed shuffle to True so that both batches and the val batches shoot up the same accuracy. So we have completed the data splitting part by using the train test split method from the sklearn library.

Table 3.2: Number of Slices in Datasets

Dataset	Training set	Test set
MS	3013	753
Healthy	80	20

The work flow of Detection of MS using Deep learning is shown in Figure 3.4

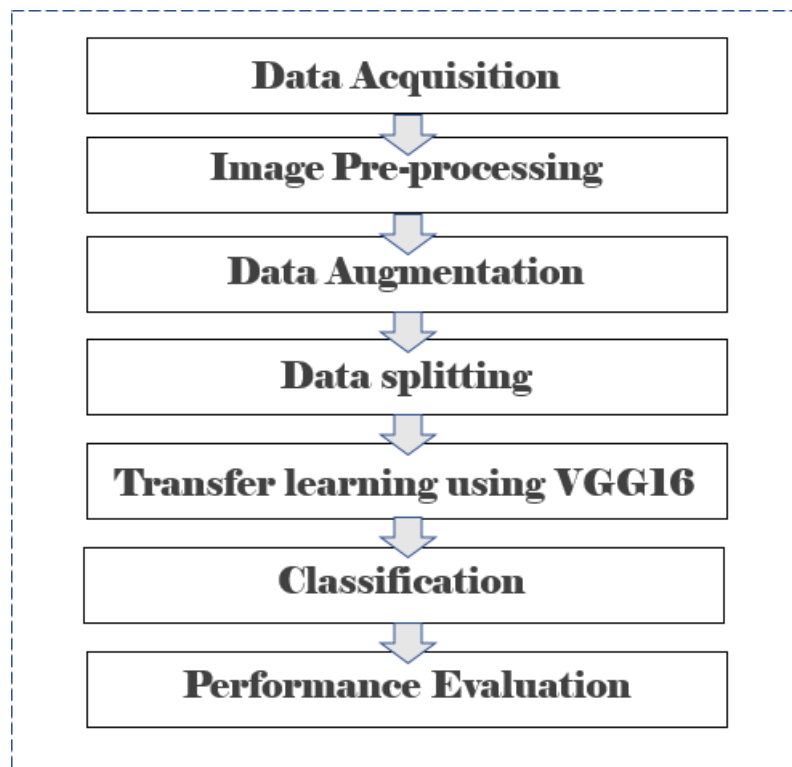


Figure 3.4: Work Flow

Chapter 4

Prediction Modeling using CNN

4.1 Convolutional Neural Network

A Convolutional Neural Network takes an image as an input and assigns different features for the images. These features are a variety of aspects in the image which helps them to differentiate from the other images. In comparison with other classification algorithms, CNN requires less pre-processing. As we already know that computer sees an image in the form of a pixel depending on the image resolution. The computer sees the image as *height width dimension* where dimension denotes the channel of the image. When the dimension is 3, it refers to RGB values and when the value is 1, it refers to a grayscale image. Practically, deep CNN models for training and testing each input image. Each input image is modeled by passing through an alternate series of convolutional layers with Kernels(filters), Pooling, Fully connected layers. And to classify the image, the softmax function is used.

Feature Learning is comprised of repeating and alternating convolution layer, rectified linear unit (ReLU) layer, and pooling layer[16]. The input of the convolution layer is fixed at 224 224 size of RGB image. Both input and filters have the same number of channels.[16].

Table 4.1: Size of Input, Filter and Output

Operator	Size
Input	$H_I \times W_I \times C$
Filter	$H_F \times W_F \times C \times Y$
Output	$H_O \times W_O \times Y$

Here, the number of filters is Y having the size of $H_F \times W_F \times C$, where H_F is the heights and W_F is the width of each filter and C is the number of channels[16]. Again, the size of the input is $H_I \times W_I \times C$, where H_I is the height and W_I , is the width of input[16](Figure 4.2).

Conventional feature learning moves the kernel from top to bottom along with the left to right direction with the stride of 1. And also convolution padding are set to 1 pixel for 3 3 convolution layers[14]. After the convolution layer, the feature map passes through a nonlinear function. We have used a rectified linear unit (ReLU) as a nonlinear function although the sigmoid function is used conventionally. The

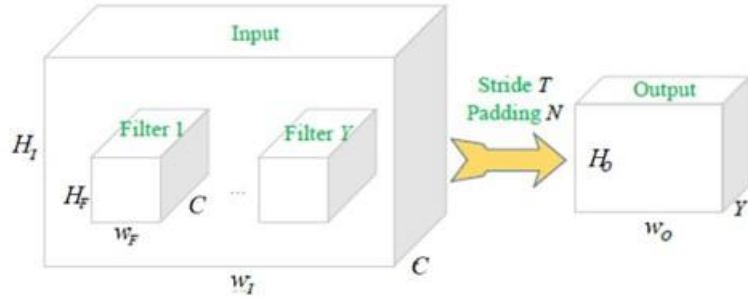


Figure 4.1: Implementation of Convolution layers

rectified linear unit (ReLU) substrates at the region where the input is negative[24]. We have used ReLU to stop forwarding the negative value through the network. The output from the ReLU layer passes through the pooling layer. The pooling layer consists of five max-pooling layers. Max-pooling is implemented here for a 2×2 pixel window having a stride set to 2[14].

In order to gain success in the fields of computer vision, the implementation of Convolutional Neural Network has become very well-known. Convolutional Neural Network is a deep neural network. Our work in this paper consists of three parts of CNN: feature learning, transfer learning and classification. A CNN model was implemented to classify brain images. In the feature Learning part, there are alternating convolutional layers and pooling layer, with Rectifier Unit (ReLU) activation function. In the Transfer Learning part, we have modified the pre-trained model for serving the purpose of our work. In the classification part, there are a fully connected layer and a softmax layer. The proposed CNN architecture illustrates in Figure 4.1. The CNN model was implemented using Keras python deep learning libraries.

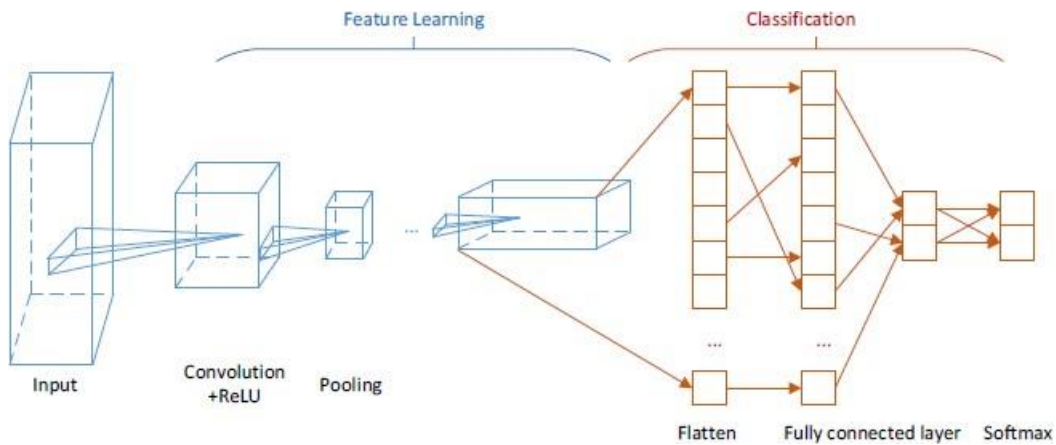


Figure 4.2: Diagram of Convolutional Neural Network

4.2 Transfer Learning

The main purpose of Transfer Learning is to take a model that is pre-trained on a large dataset and transfer the knowledge to a smaller dataset[26]. The convolutional layers extract the low-end features which are applicable across images like edges, patterns, gradients[26]. And the later layers identify specific features present in an image[26]. Convolutional Neural Networks are unable to learn properly when the training data is insufficient. Transfer learning has a significant contribution to solving this issue. Transfer learning adjusts the model in such a way that the new task with a smaller dataset can be completed. In the detection of MS lesions from MRI scans, we are using the VGG16 model for transfer learning. The sample images we could extract from the dataset is not enough for building a classifier. The VGG-16 model consists of 16 layers. VGG16 can be easily used for feature extraction as it is pre-trained on millions of images. After data preparation, we extracted the features from the VGG-16 model.

block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

Total params: 14,714,688		
Trainable params: 14,714,688		
Non-trainable params: 0		

Figure 4.3: Before Transfer Learning

Now, after feature learning, we set include_top to False on the pre-trained VGG16 so that the Fully Connected head is not loaded[22]. Because we don't want to load the Fully connected output layers of the model to make any prediction. After chopping off the head we specify an input of size 224 224 3 same as the size of the input image of the original VGG16 model[22]. We are doing this to add a new output layer and to further train it. And then we pass the output through the average pooling layer. We do this so that the global average pooling is applied to the output of the last fully connected block. As a result, the output of the model will be in a 2D tensor. The output of the average pooling layer is flattened at a 1D vector. The output flattened 1D vector is now passed through a dense layer after passing it through the non-linear activation function called ReLU. We pass the output through the dense layer to connect every input to every output by weight of $n \text{ inputs } n \text{ outputs}$ weights. Moreover, dropout is applied in between the two hidden layers and before the output layer. 20% is used as dropout rate as a weight

- × -

constraint on those layers[9]. After passing through the dense layer, we use the softMax activation function to normalize the output. Finally, we set the base layers to not trainable in the VGG16 model so that the base layer does not update during the backpropagation phase[22].

block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d_1 (Average)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense_3 (Dense)	(None, 64)	32832
dropout_2 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 30)	1950
dropout_3 (Dropout)	(None, 30)	0
dense_5 (Dense)	(None, 5)	155
=====		
Total params: 14,749,625		
Trainable params: 34,937		
Non-trainable params: 14,714,688		
=====		
None		

Figure 4.4: After Transfer Learning

4.3 Classification

The yield of the transfer learning is flattened at a 1D vector and further passed through the classification phase. We use fully-connected layers in our model. The neurons in a fully-connected layer have a connection with all the previous layers. As a result, many parameters are produced in the layer. The last fully-connected layer will have the same number of neurons as the number of classes. And the last fully connected layer will also output the score of each class as shown in Figure4.5.

We use softmax activation at the end with five units of the dense layer as we have five classes for prediction. The five classes are RR, PR, SP, CIS, and Healthy. The softmax layer will output values like 10000, 01000, 00100, 00010, 00001 based on the confidence of the model to which class the image belongs to[24].

4.4 Model Compilation

After the model is prepared, we have compiled the model. In the aim of reaching the global minima while training the model, Adam Optimiser has been used[24]. In case we are stuck in local minima Adam optimizer will assist us to get to the global minima. We have set the learning rate of the optimizer at $1e^{-3}$. In the process, we have also noticed of the training is bouncing a lot on epochs. In case, it bounces

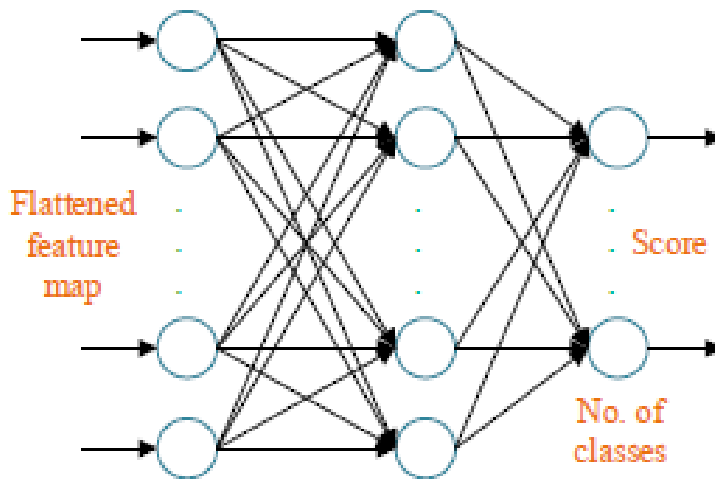


Figure 4.5: Diagram of fully-connected layer

we had to decrease the learning rate to reach the global minimum. The summary of the model is shown in figure 4.6.

After compiling we have used model.fit generator to proceed the data to the model. We passed both the training data and test data to fit generator[24]. We have set the value of steps per epoch to the length of training data and validation steps to the length of test data. And it will set the batch size to the model.

To make a prediction on the trained model we have loaded the best-saved model and pre-processed the image and passed the image through the model for output.

Model: "model_1"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d_1 (Average Pooling2D)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense_3 (Dense)	(None, 64)	32832
dropout_2 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 530)	1950
dropout_3 (Dropout)	(None, 30)	0
dense_5 (Dense)	(None, 5)	155
Total params: 14,749,625		
Trainable params: 34,937		
Non-trainable params: 14,714,688		

Figure 4.6: Summary of the model

Chapter 5

Result and Analysis

5.1 Comparison

While working with our approach, we have also tried to learn about other different approaches to detect MS lesions from MRIs. After completion of the implementation of our approach, we have compared the accuracy of our approach and the other approaches we have learned about. Below we have shown the studies on MS lesions from MRIs having different accuracies in Table 5.1.

Table 5.1: The obtained results from different top rank strategies

Method	Sensitivity	Specificity	Accuracy
Proposed	100%	95.45%	98.24%
Sepahvand et. al[28]	69%	97%	95%
Doğan et. al[19]	–	–	70.24-91.04%
Narayana et. al[21]	78±4.3%	73±2.7%	75%

So from the table above we can state that our approach is acceptable as we obtained better accuracy than the proposed approaches in these studies.

We have tried different other kinds of CNN techniques to see if it gives any better accuracy. The two other approaches we implemented to detect MS lesions are VGG19 and Resnet50. VGG19 and Resnet50 both fall under the CNN technique. From the three approaches, Resnet50 is the fastest method but with the lowest accuracy. Resnet50 takes 244s per epoch on average. And VGG19 is the slowest method with moderate accuracy. VGG19 takes 490s per epoch on average. Though VGG16 is not the fastest method it gives the highest accuracy among the three. VGG16 takes 390s per epoch on average. The Detection result using different approaches are shown in Table 5.2.

Table 5.2: The obtained results from different strategies of CNN

Method	Sensitivity	Specificity	Accuracy
VGG16(proposed)	100%	95.45%	98.24%
VGG19	100%	99%	95.96%
Resnet50	100%	99%	79.92%

5.2 Performance Efficiency

The results were evaluated using a confusion matrix which is a part of the sklearn library. We have used a confusion matrix to get a summary of the prediction results on our categorical classification problem. A confusion matrix summarizes the number of correct and incorrect predictions with count values separated by each class (Figure 5.2). It says that the confusion matrix shows the way when your classification model is confused while making predictions. Before calculating a confusion matrix we made a prediction for each row in the testing set using the model.predict function. We need the function to know the outcome classes for the testing set. After making the prediction, we can get the number of correct predictions for each class. The number of incorrect predictions for each class can also be predicted or- ganized by the class. We have used the confusion matrix for five class values.

```
[INFO] evaluating network...
                precision    recall  f1-score   support

/Users/17101129/Downloads/dataset/train_images/CIS      0.98      0.98      0.98      126
/Users/17101129/Downloads/dataset/train_images/NO      1.00      0.88      0.93       24
/Users/17101129/Downloads/dataset/train_images/PR      1.00      0.88      0.93       73
/Users/17101129/Downloads/dataset/train_images/RR      0.98      1.00      0.99      621
/Users/17101129/Downloads/dataset/train_images/SP      1.00      0.99      1.00      122

                accuracy
                macro avg      0.99      0.94      0.97      966
                weighted avg    0.98      0.98      0.98      966

accuracy: 0.9824
sensitivity: 1.0000
specificity: 0.9545
precision: 1.0000
Recall: 0.9545
```

Figure 5.1: Detection Results from VGG16 model

Using VGG16 we could achieve the highest accuracy among other CNN techniques. The accuracy we obtained using VGG16 is 98.24%. Overall, we have obtained a sensitivity of 100%, a specificity of 95.45%, and precision of 100%.

Then we can visualize the training accuracy and loss using the matplotlib library (Figure 5.2)[24].

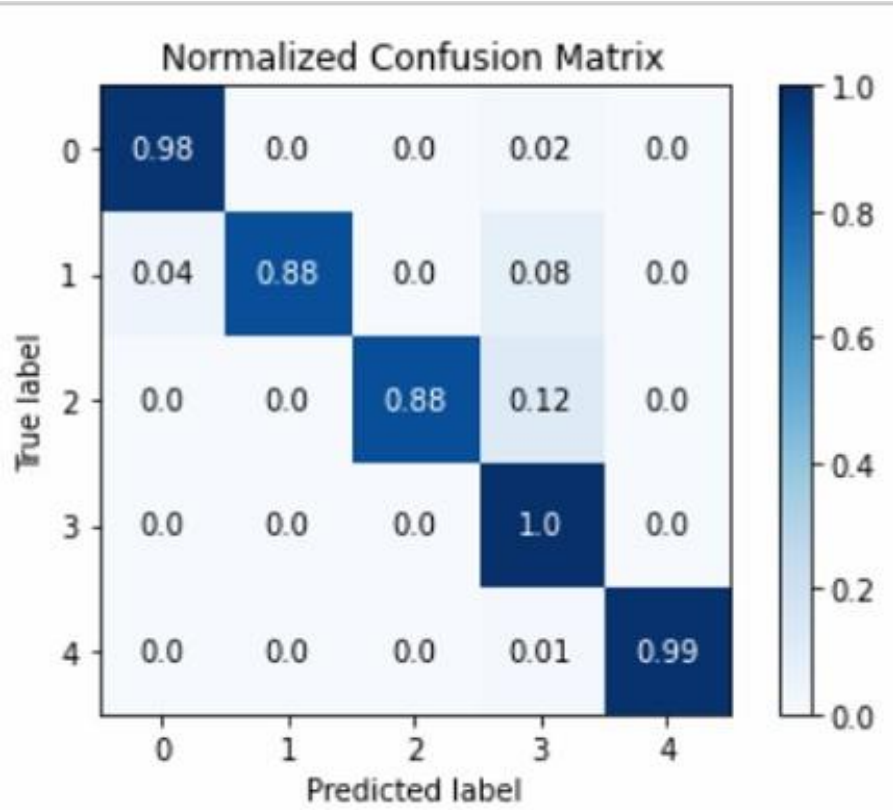


Figure 5.2: Normalized Confusion Matrix

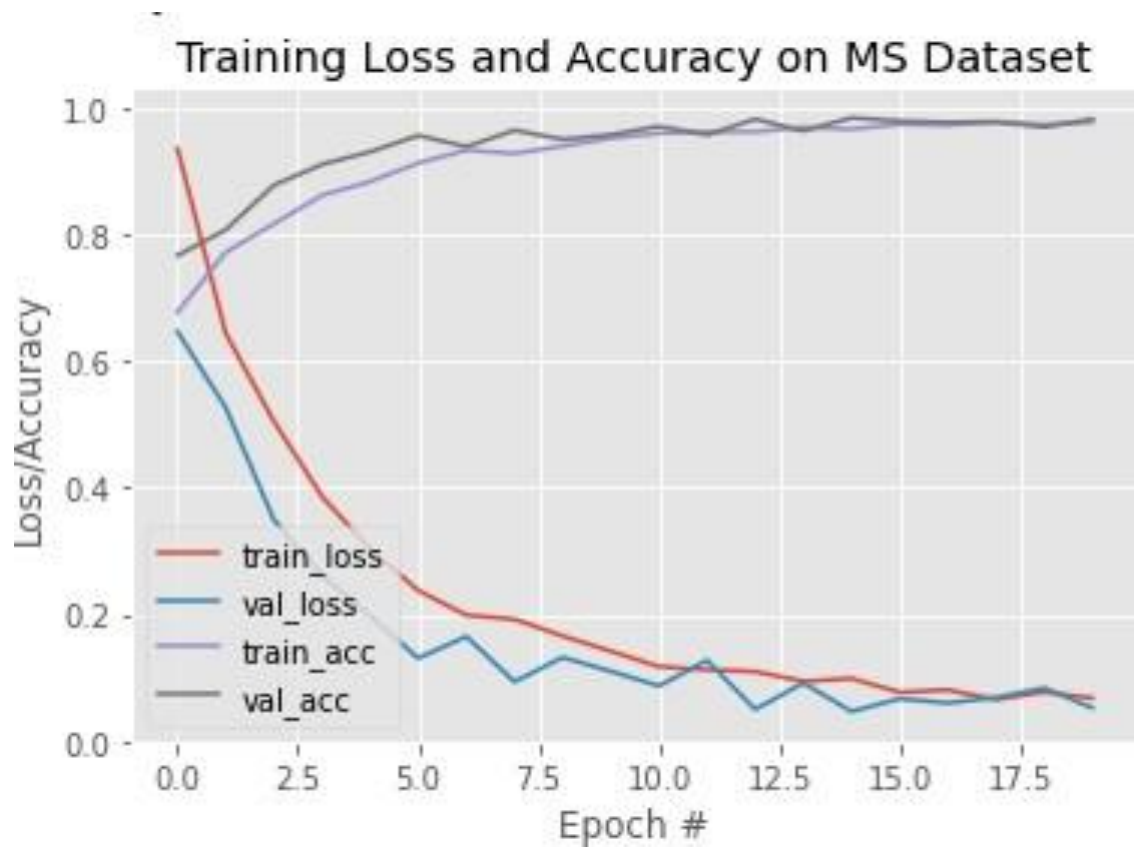


Figure 5.3: Training loss and accuracy curve

Chapter 6

Conclusion and Future Work

In this paper, we have shown that a deep learning framework can reliably and efficiently detect MS lesions from MRI scans and classify the progression of the disease. The model we have used here is VGG16, which is a part of CNN called transfer learning. As the purpose of Transfer learning is to transfer the knowledge to a smaller dataset after training a model on a large dataset, we could efficiently work on the smaller dataset. To obtain a more accurate result, we have also considered a dataset of healthy MRI scans. And our approach aims to accelerate the automatic detection and classification of Multiple Sclerosis disease. The flair-attenuated inversion recovery (FLAIR) MRI scans assists the system and optimize the total time of execution. Moreover, the overall performance of the system shoots an accuracy of 98.24%

In the future, we aim to increase the accuracy minimizing the execution time. We also aim to use a larger dataset so that we can obtain a robust system with high efficiency and accuracy in a minimum time of execution. Moreover, we also plan to conduct the system in such a way that we can make it computationally lighter. By making it computationally lighter, it can also be implemented on smaller devices, for instance, mobile phones, tablets. In the present world, we rely on small devices as it provides us with more convenience and portability. So such an approach will surely add exemplary value to our work.

In conclusion, the obtained result indicates that the proposed model can reliably and accurately provide us with the detection of MS lesions and also classifying the progression of the MS lesions.

Bibliography

- [1]K. Van Leemput, F. Maes, D. Vandermeulen, A. Colchester, and P. Suetens, “Automatic segmentation of multiple sclerosis lesions by model outlier detection,” *IEEE Transactions on Medical Imaging*, vol. 20, pp. 677–688, Aug. 2001.
- [2]O. Pampliega, M. Domercq, P. Villoslada, J. Sepulcre, A. Rodríguez-Antigüedad, and C. Matute, “Association of an *eat2* polymorphism with higher glutamate concentration in relapsing multiple sclerosis,” *Journal of neuroimmunology*, vol. 195, pp. 194–8, Apr. 2008. DOI:10.1016/j.jneuroim.2008.01.011.
- [3]A. Akselrod-Ballin, M. Galun, J. M. Gomori, M. Filippi, P. Valsasina, R. Basri, and A. Brandt, “Automatic segmentation and classification of multiple sclerosis in multichannel mri,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 10, pp. 2461–2469, 2009. DOI:10.1109/TBME.2008.926671.
- [4]J. Liu, C. D. Smith, and H. Chebrolu, “Automatic multiple sclerosis detection based on integrated square estimation,” in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2009, pp. 31–38. DOI:10.1109/CVPRW.2009.5204351.
- [5]S. Jain, D. Sima, A. Ribbens, M. Cambron, A. Maertens, W. Van Hecke, J. De Mey, F. Barkhof, M. Steenwijk, M. Daams, F. Maes, S. Huffel, H. Vrenken, and D. Smeets, “Automatic segmentation and volumetry of multiple sclerosis brain lesions from mr images,” *NeuroImage: Clinical*, vol. 32, May 2015. DOI: 10.1016/j.nicl.2015.05.003.
- [6]Z. Zeng, S. Chen, L. Yin, and R. Zwiggelaar, “Unsupervised segmentation for multiple sclerosis lesions in multimodality magnetic resonance images,” in *2015 8th International Conference on Biomedical Engineering and Informatics (BMEI)*, 2015, pp. 126–130. DOI:10.1109/BMEI.2015.7401486.
- [7]S. M. Ali and A. Maher, “Identifying multiple sclerosis lesions in mr images using image processing techniques,” in *2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)*, 2016, pp. 1–5. DOI:10.1109/AIC-MITCSA.2016.7759913.
- [8]T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. Li, A. Traboulsee, and R. Tam, “Deep 3d convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1229–1239, 2016. DOI:10.1109/TMI.2016.2528821.

- [9]J. Brownlee, “Dropout regularization in deep learning models with keras,” 2016. [Online]. Available:<https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/?fbclid=IwAR2uigLvHZSxh8bfbsf5JRME1hTV09KS5e5ygYs3cLKC7BqMXued-oL17A>.
- [10]N. Gunawardena, R. Rajapakse, and N. Kodikara, “Applying convolutional neural networks for pre-detection of alzheimer’s disease from structural mri data,” Nov. 2017, pp. 1–7. DOI:10.1109/M2VIP.2017.8211486.
- [11]S. P. Washimkar and S. D. Chede, “Prediction of multiple sclerosis in brain mri images using hybrid segmentation,” in *2017 International Conference on Signal Processing and Communication (ICSPC)*, 2017, pp. 234–239. DOI:10.1109/CSPC.2017.8305845.
- [12]M. Alshayegi, M. Al-Rousan, h. ellethy hanem, and S. Abed, “An efficient multiple sclerosis segmentation and detection system using neural networks,” *Computers Electrical Engineering*, vol. 71, pp. 191–205, Oct. 2018. DOI:10.1016/j.compeleceng.2018.07.020.
- [13]S. Aslani, M. Dayan, L. Storelli, M. Filippi, V. Murino, M. Rocca, and D. Sona, *Multi-branch convolutional neural network for multiple sclerosis lesion segmentation*, Nov. 2018.
- [14]Neurohive, “Vgg16 – convolutional network for classification and detection,” 2018. [Online]. Available:<https://neurohive.io/en/popular-networks/vgg16/?fbclid=IwAR1xMkE6tiGcVVRVYXdOFYEWO3RARITV5iaWks3Xt9buxudt9YIYHesgn0>.
- [15]Y. Samah, B. Salem Yassine, and A. Mohamed Naceur, “Multiple sclerosis lesions detection from noisy magnetic resonance brain images tissue,” in *2018 15th International Multi-Conference on Systems, Signals Devices (SSD)*, 2018, pp. 240–245. DOI:10.1109/SSD.2018.8570679.
- [16]Y.-D. Zhang, C. Pan, J. Sun, and C. Tang, “Multiple sclerosis identification by convolutional neural network with dropout and parametric relu,” *Journal of Computational Science*, vol. 28, Jul. 2018. DOI:10.1016/j.jocs.2018.07.003.
- [17]Q. Zhou and X. Shen, “Multiple sclerosis identification by grey-level cooc-currence matrix and biogeography-based optimization,” in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, 2018, pp. 1–5. DOI:10.1109/ICDSP.2018.8631873.
- [18]N. Chakrabarty, *Brain mri images for brain tumor detection*, Apr. 2019.
- [19]A. F. Doğan and D. G. Duru, “Comparison of machine learning techniques on ms lesion segmentation,” in *2019 Medical Technologies Congress (TIPTE-KNO)*, 2019, pp. 1–4. DOI:10.1109/TIPTEKNO.2019.8895202.
- [20]E. Kats, J. Goldberger, and H. Greenspan, “Soft labeling by distilling anatom- ical knowledge for improved ms lesion segmentation,” in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, 2019, pp. 1563–1566. DOI:10.1109/ISBI.2019.8759518.
- [21]P. Narayana, I. Coronado, S. Sujit, J. Wolinsky, F. Lublin, and R. Gabr, “Deep learning for predicting enhancing lesions in multiple sclerosis from non- contrast mri,” *Radiology*, vol. 294, p. 191061, Dec. 2019. DOI:10.1148/radiol.2019191061.

- [22]A. Rosebrock, “Change input shape dimensions for fine-tuning with keras,” 2019. [Online]. Available:https://www.pyimagesearch.com/2019/06/24/change-input-shape-dimensions-for-fine-tuning-with-keras/?fbclid=IwAR3qq03r7sCbIZP3QucNsgQM9N_qmXb5sGEh00.
- [23]M. Salem, S. Valverde, M. Cabezas, D. Pareto, A. Oliver, J. Salvi, À Rovira, and X. Lladó, “Multiple sclerosis lesion synthesis in mri using an encoder- decoder u-net,” *IEEE Access*, vol. 7, pp. 25 171–25 184, 2019. DOI:10.1109/ACCESS.2019.2900198.
- [24]R. Thakur, “Step by step vgg16 implementation in keras for beginners,” 2019. [Online]. Available:<https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c>.
- [25]F. La Rosa, A. Abdulkadir, M. Fartaria, R. Rahmzadeh, P.-J. Lu, R. Gal- busera, M. Baraković, J.-P. Thiran, C. Granziera, and M. Cuadra, “Multi- ple sclerosis cortical and wm lesion segmentation at 3t mri: A deep learning method based on flair and mp2rage,” *NeuroImage: Clinical*, vol. 27, p. 102 335, Jun. 2020. DOI:10.1016/j.nicl.2020.102335.
- [26]M. Masum, “A poor example of transfer learning: Applying vgg pre-trained model with keras,” 2020. [Online]. Available:<https://towardsdatascience.com/a-demonstration-of-transfer-learning-of-vgg-convolutional-neural-network-pre-trained-model-with-c9f5b8b1ab0a>.
- [27]U. N. L. of Medicine, “Multiple sclerosis,” 2020. [Online]. Available:<https://medlineplus.gov/ency/article/000737.htm?fbclid=IwAR1AN5Cs3GDVIU2JrGiL9rEy3rtk0ruQOPrCmK8QpKDXmL18YDU6IX97k0>.
- [28]N. M. Sepahvand, D. L. Arnold, and T. Arbel, “Cnn detection of new and enlarging multiple sclerosis lesions from longitudinal mri using subtraction images,” in *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 2020, pp. 127–130. DOI:10.1109/ISBI45749.2020.9098554.
- [29]N. M. S. Society, “Types of ms,” 2020, EIN: 13-5661935. [Online]. Available: <https://www.nationalmssociety.org/What-is-MS/Types-of-MS>.
- [30] *3d mr image database of multiple sclerosis patients with white matter lesion segmentations*, Trzaska cesta 25, SI-1000 Ljubljana, Slovenia.