# Multiple Sclerosis Lesion Detection via Machine Learning Algorithm based on Converting 3D to 2D MRI Images by Using Value of Binary Pattern Classification and Computational Methods

Thesis for the Degree of Doctor of Philosophy (PhD)

by **Moghadasi Mohammad**

Supervisor: **Dr. Fazekas Gábor**

UNIVERSITY OF DEBRECEN

Doctoral Council of Information Technology
Doctoral School of Informatics

Debrecen, 2021

***To my beautiful family,***

*Hereby I declare that I prepared this thesis within the Doctoral Council of Natural Sciences and Information Technology, Doctoral School of Informatics, University of Debrecen in order to obtain a PhD Degree in Informatics at Debrecen University.*

*The results published in the thesis are not reported in any other PhD theses.*

*Debrecen, 2020. December* *signature of the candidate*

*Hereby I confirm that Moghadasi Mohammad, candidate conducted his studies with my supervision within the Informatics, Doctoral Program of the Doctoral School of University of Debrecen, between 2012 and 2017. The independent studies and research work of the candidate significantly contributed to the results published in the thesis.*

*I also declare that the results published in the thesis are not reported in any other theses.*

*I support the acceptance of the thesis.*

*Debrecen, 2020. December* *signature of the supervisor*

# MS lesion detection via machine learning algorithm based on converting 3D to 2D MRI images by using value of binary pattern classification and computational methods

Dissertation submitted in partial fulfilment of the requirements
for the doctoral (PhD) degree in Informatics

Written by Moghadasi Mohammad certified in Informatics

in the framework of the Informatics doctoral school of the University of Debrecen
(Discrete Mathematics, Image processing and visualization programme)

Dissertation advisor: Dr. Fazekas Gábor.

The comprehensive examination board:
chairperson: Prof. Vaszil György
members: Dr. Zichar Marianna
Dr. Palágyi Kálmán

The date of the comprehensive examination: 16th November 2017.

The official opponents of the dissertation:
Dr. ..................................................... ............
Dr. ..................................................... ............
Dr. ..................................................... ............

The evaluation committee:

chairperson: Dr. ..................................................... ............
members: Dr. ..................................................... ............
Dr. ..................................................... ............
Dr. ..................................................... ............
Dr. ..................................................... ............

The date of the dissertation defense: …………………………… 2021

# Acknowledgment

Firstly, I would like to acknowledge my indebtedness and render my warmest thanks to my supervisor, Dr. Fazekas Gábor, who made this work possible, for his immense knowledge, motivation, patience and continuous support through my Ph.D. studies. His friendly guidance and expert advice have been invaluable throughout all stages of the work.

The dissertation has also benefited from insightful and wise comments made by Prof. Fazekas Attila and Dr. Remzső Tibor, who have read through the manuscript and also for the hard questions which motivated me to widen my research from various perspectives, therefore I’d like to take this opportunity to thank them.

Besides my advisors, I would like to acknowledge and express my gratitude to the rest of my thesis committee and the members: Prof. Pethő Attila, Prof. Vaszil György, Prof. Terdik György, Prof. Hajdu András, Prof. János Sztrik, Dr. Andrea Pintér-Huszti, Dr. Zichar Marianna, Dr. Palágyi Kálmán, Dr. László Kozma, Dr. Kocsis Gergely and also everyone who participated.

Special thanks to Dr. Majid Mousavi, Viski Annamária, Kun Dávid, Nicholas Nocheck, Ethan Roth and Kovács Enikő.

Last but not least, exceptional thanks are due to my family, my parents, my brothers and my sister for supporting me spiritually during the course of writing this thesis and for their continuous support and understanding throughout my life in general.

# Abstract

In the twenty first century, there have been various scientific discoveries which have helped in addressing some of the fundamental health issues. Specifically, the discovery of machines which are able to assess the internal conditions of individuals, has been a significant boost in the medical field. Multiple Sclerosis (MS) is a demyelinating disease in which the insulating covers of nerve cells in the brain and spinal cord are damaged. This damage disrupts the ability of the nervous system’s parts in various forms to transmit signals, therefore results a wide range of symptoms including physical, mental and psychiatric problems. In this regard, most of the researchers have focused on improving the classification of brain lesions, especially those from MS, which is a relatively difficult task, using different algorithms, but mainly in determining the edge. However, the absence of learning methods have caused complexity and difficulties for having accurate results in the previous works. Therefore, this need motivated us to use various tools of learning methods with a focus on Cellular Learning Automata (CLA) to achieve more accurate results in detection of MS lesions.

Cellular Learning Automata (CLA) is a hybrid model of two, Learning Automata and Cellular Automata, which is a simple discrete system that can exhibit complex calculations and behavior through simple and local rules. In this study, we aim to propose a new combinational algorithm using Support Vector Machine (SVM) used for classification and cellular learning automata (CLA) to increase the accuracy of MS lesion detection. The objective is to create artificial models using support vector machines (SVM) to classify MS and normal brain MRI images, analyze the effectiveness of these models and their potential to use them in multiple sclerosis (MS) diagnosis. In order to develop such combination method, we start with simple learning methods such as k-means to find MS lesion.

The research was carried out in four stages, respectively, the algorithms are as follows; a) Semi-automatic method and use of K-Means, b) Automatic MS Segmentation Approach Based on Cellular Learning Automata, c) MS Segmentation Approach based on SVM, CLA and K-Means, d) Accurate Simulated Database, 3D MRI to 2D Images, using value of Binary Pattern Classification for MS Detection. The algorithms consist of pre-processing parts, detecting MS-hemispheres, feature extraction, classification and post-processing. In the pre-processing section, the brightness intensity of the normalized images and the brain region are first extracted. Then, to reduce the computational volume, the lesions are diagnosed. The proposed approach can be considered as a supplementary or superior method for other methods such as Graph Cuts (GC), fuzzy c-means, mean-shift, k-Nearest Neighbor (KNN). We try to see the benefits of having a 3D database but to use 2D vectors only for better comparison and more accurate results. Support vector machines (SVM) can be a useful tool during the multiple sclerosis (MS) disease diagnosis process, however to be able to make better assumptions, more tests are needed.

# Contents

**Acknowledgment**

**Abstract**

**Contents**

**List of Figures**

**List of Tables**

# List of Figures

# List of Tables

# Chapter 1

## Introduction

## 1-1- Introduction

One of the most common autoimmune diseases of the central nervous system is multiple sclerosis (MS), which mainly affects young people from the age of 6 and causes problems such as loss of control, vision, balance and senses. Autoimmune diseases are diseases in which the immune system, which is responsible for protecting the body and destroying external damages such as bacteria, mistakenly begins to invade the body's tissues. In MS, the immune system begins to invade the brain and spinal cord, the two major components that control the central nervous system, causing paralysis and disability. This is why it is known as an autoimmune disease [1].

The central nervous system is made up of millions of axons, which are responsible for the rapid transmission of nerve signals from one neuron to another in order to transmit the message between the brain and the body. Axons are covered with a protective layer called myelin (a type of fat) that prevents the interference of nerve signals [2]. In MS, the immune system can disrupt the transmission of nerve signals by axons and the brain loses control over the body by disrupting these fats and destroying them. Depending on the affected area, different symptoms may appear such as partial or complete loss of vision usually in one eye, pain in moving the eyes and vision or blurred vision, severe muscle weakness and destruction, weakness or numbness of one or more parts of the body, body aches, tingling or pain in some parts of the body, electrical shock when certain head movements, body shaking or body irregularities, stuttering, fatigue and dizziness. In addition to damaging the axons, MS causes numerous inflammatory foci that are reversible despite axonal death. Therefore, axonal mortality is considered as an appropriate marker for disease progression. People may occasionally feel these symptoms at the onset of recurrence and then they may temporarily disappear. Some people have a benign form of the disease and do not develop another attack after the initial attack, but only in some cases and for people with problems such as dryness or cramping, paralysis, especially paralysis of the legs, bladder and abdominal problems, mental problems such as forgetfulness or lack of concentration, depression and epilepsy or anesthesia [3].

MS patients are classified into four clinical groups [4]: Relapsing-Remitting MS (RRMS) which is the most common form, 85% of MS patients. Progressive-Relapsing MS (PRMS) which is a rare form of MS and only 5%. Primary Progressive MS (PPMS) and Secondary Progressive MS (SPMS), which are not very common, 10% of patients. The RRMS type is the most common type of disease associated with recurrence. The return of RRMS is linked with a partial or complete recovery in a few weeks or months. In SPMS, symptoms worsen more steadily over time and it also includes a period of sustained improvement. SPMS patients with RRMS have different degrees of disability and no recovery. Patients with PPMS have been progressing since the onset of the disease and in some cases have improved slightly.

Although 10% of patients with MS recover within 20 years, 50% of them need walking assistance after 15 years and in some cases die. Studies on the association between onset of disease and age show that the younger the age of the disease, the more MS lesions occur and the more part of the brain tissue (especially the white area of the brain) is analyzed. Although MS was discovered in 1868, its cause remains largely unknown. Researchers know that MS is a nerve damage caused by inflammation, but the cause of the inflammation is still unknown. However, the effects of environmental and genetic factors have been reported as the most common causes of the disease. Ages 20 to 40, having a family member, infections and specific viruses such as Einstein Barr virus, white people, especially families living in northern Europe, developing other autoimmune diseases such as thyroid disease, type 1 diabetes or Inflammatory bowel disease increase the risk of MS [1].

After initial examination, MRI imaging can confirm the diagnosis. In MRI images, doctors can see where the plaques originate and can even estimate how long they have been formed. Although about 5% of MS patients have normal MRI images at the onset of their disease, and interpretation of these images should always be accompanied by other symptoms and clinical examination, such as spinal cord fluid testing and neuropsychological tests, but in 95% of cases MRI plaques are visible on MRI images. So given the high percentage of diagnoses seen in these images, MRI is currently the most accurate method that can help diagnose MS [5].

## 1-2- Problem Statement and Motivation

As mentioned in the introduction, MS is a disorder of the central nervous system autoimmunity that the immune cells mistakenly begin to destroy the protective lipid sheath on the nerve fibers. These affected areas are called plaque in pathology [6]. The distribution of these plaques is white in body tissue. As the process of repair and destruction of the protective sheath on the nerve fibers continues, the messaging system becomes disrupted and causes symptoms such as fatigue, limb weakness, disorganization, dizziness, and visual disturbances [7]. Currently the best way to diagnose MS disease is by MRI imaging of the brain and neck at different intervals [8]. These lesions are visible in white matter in MRI images, appearing in T2W[1] images with high light intensity and appearing in T1W[2] images with less light intensity. It is very difficult to diagnose MS because of the location, size, shape, and anatomy of the individual. In lesion segmentation, there are two methods of automatic and semi-automatic. In semi-automatic methods the expert can change the segmentation until satisfied. The goal of semi-automatic techniques is to reduce processing time and reduce the diversity of experts, and the goal of automated methods is to provide more accurate segmentation [9]. In binary segmentation, four scenarios may occur: TPs[3], TNs[4], FPs[5], FNs[6]. In order to validate the automated segmentation method, the accuracy and reproducibility criteria are used, which uses mean and standard deviation and, also to evaluate the accuracy of the three criteria of lesion load, overlap, and lesion-based. The most widely used criterion for segmentation is the similarity index or DSC[7]. Criteria includes Sensitivity, Specificity and Accuracy [10]. Most of the researchers have focused on improving the classification of brain lesions, especially those from MS,

---

[1] T2-Weighted
[2] T1-Weighted
[3] True Positive
[4] True Negative
[5] False Positive
[6] False Negative
[7] Dice Similarity Coefficient

which is a relatively difficult task, using different algorithms, but mainly in determining the edge. Since the absence of learning methods have caused complexity and difficulties for having accurate results in the previous works. Therefore, this need motivated us to use various tools of learning methods with a focus on Cellular Learning Automata (CLA).

CLA is a hybrid model of two, Learning Automata and Cellular Automata which is a simple discrete system that can exhibit complex calculations and behavior through simple and local rules. In this study, we aim to propose a new combinational algorithm using Support Vector Machine (SVM) used for classification and Cellular Learning Automata (CLA) to increase the accuracy of MS lesion detection. The objective is to create artificial models using support vector machines (SVM) to classify MS and normal brain MRI images, analyze the effectiveness of these models and their potential to use them in Multiple Sclerosis (MS) diagnosis. In order to develop such combination method, we start with simple learning methods such as k-means to find MS lesion.

## 1-3- Novelty and Contributions

In Reference [11], many previous methods use prior knowledge of voxels classification, thus preventing the discovery of other different texture classifications from classes taught to the system. Because of this,

this paper proposes two unsupervised methods for segmenting the brain image. The first method is based on the use of relevant information extracted from the volume and total histograms processed using self-organization maps (SOM). The second method consists of four steps, the first being the acquisition of MRI images, the second is a feature extraction procedure using overlapping windows, the third is a dynamic-based feature selection, and the last, grouping the map units using the SOM clustering algorithm capability. Reference [12] presents a new genetic algorithm-based software that derives a CNN (Cellular Neural Network) that is capable of automatically determining the lesion burden in MS patients from MRI images. In [13], the author reviews gray matter leanness that is highly correlated with physical disability and clinical course in patients with MS. There are univariate and multivariate patterns in this area. Univariate patterns based on single voxels detect group differences or clusters. In the context of multivariate models, SVMs (Support Vector Machine) are used as a powerful diagnostic tool because of their ability to learn complex classification, high dimensionality of training data, and extend classification rules to latent data. SVMs use information from all available voxels that have been combined to reflect differences between groups [14]. More recently, SVMs have been used to scan MRI images of MS patients, to identify patterns in brain regions that are apparently normal using an ROI (Regions of Interest) method that demonstrates MS patients that can be controlled. Light intensity patterns based on local brain tissue are separated from healthy individuals. Reference [15] provides a method in which sequential excisions are segmented together to provide a temporally adapted tissue segmentation as long as they are sensitive to the emergence of new lesions. Reference [16] provides a fully automated method for segmenting brain images containing white matter lesions. This method is an atlas-based segmentation technique that uses topological atlas as well as statistical atlas. Reference [17] presents a hierarchical segmentation-based and self-organization map (SOM) segmentation approach and also multipurpose-based feature selection to optimize segmentation process performance. Reference [18] presents a fully automatic method that uses the information of all T1W, T2W, FLAIR and PD images. This method is based on the K-Nearest Neighbor (KNN) classification techniques that create a characteristic space of voxels' light intensity and three-dimensional spatial information.

## 1-4- Objective

This research was carried out in four objectives as followings:

a) How to integrate semi-automatic method and K-Means for detecting MS lesions in MRI images.

b) How to use Cellular Learning Automata (CLA) for automatic MS Segmentation.

c) How to integrate SVM, K-Means and CLA to increase the accuracy of segmentation of MS lesions.

d) How to use value of Binary Pattern Classification to accurate MS Detection from Simulated 3D MRI to 2D Images.

## 1-5- Organization

This dissertation consists of six chapters:

Chapter 1- Introduction: This chapter deals with the introduction of the thesis work, motivation, and the objective of the thesis work.

Chapter 2- Theoretical Foundation: This chapter explains the basic method of MRI scanning of brain MRI images (2D & 3D) and different Sequences.

Chapter 3- Background and Related Work: This chapter describes the integrated methods of Supervised & Unsupervised learning method with CLA, Semi-automatic method integrated with K-

Means and Binary Pattern Classification to MS detection from Simulated 3D MRI to 2D Images.

Chapter 4- Proposed Method: This chapter presents the data collected over several simulation and proposes the method of the dissertation.

Chapter 5- Results and Discussion: This chapter discusses the results and the analysis.

Chapter 6- Conclusion, Suggestions and Future Works: This chapter concludes the thesis with collected results and discusses shortcoming.

# Chapter 2

## Theoretical Foundation

## 2-1- Brain MRI images

MRI is one of the successful technologies in the field of medical imaging that has made significant improvements in the quality and speed of imaging in recent years. The waves used in MRI are radio and magnetic waves that do not harm the body. In this way the patient is placed in a very strong magnetic field. This field causes the protons to rotate round that axis (known as precession), so that any proton at any moment in time will be pointing in some direction in the XY plane. These are the protons of atoms in all body tissues (especially the protons in the nucleus of the water molecule) along the lines of the MRI magnetic field. Certain radio waves are then transmitted to the patient's body. These pulsed waves cause the axis of the protons to rotate slightly. At the end of the radio pulse, the axis of the proton rotates again along the lines of the magnetic field. This reversal generates a new radio wave (electromagnetic) [19].

Secondary radio waves emitted from individual protons are then received by the MRI receivers and transmitted to the computer. The received waves are quickly analyzed and then images are made based on these analyses. An MRI computer identifies what areas of the body have emitted the most radio waves. The higher the intensity of the incoming wave, the higher the proton density at that point and because the most abundant atom in the body is the hydrogen atom in the water molecule, so it has more water wherever more radio waves are sent back. MRI can be used to lower the left (coronal), the upper (axial) and even in the direction of oblique imaging [20].

The signal strength generated by MRI depends on the two factors of proton density which are T1 and T2 relaxation times. T1 is the time when 63% of the longitudinal magnetic moment of a proton, after excitation, returns from perpendicular to the field parallel to the magnetic field. T2 is the period or time when the transverse magnetic moment of a proton drops to 37% of its initial value after excitation. In each context this time period is different. For example, at 1.5 Tesla, the T1 constant is 260 milliseconds for fat tissue and 920 milliseconds for gray matter. Most pathological processes increase their T1 and T2 relaxation time and, therefore, have a lower (darker) signal in T1-weighted images and higher signal in T2-Wighted images compared to surrounding natural tissues. Depending on what type of pulse sequence is selected, and how parameters such as repetition time (TE) and reflection time (TR) are determined (Fig. 2-1), the contrast with T1 and T2 can be illustrated and the ability of MRI to be determined. This is the special property. For example, in one image the fat is light and the other is dark [21, 22].

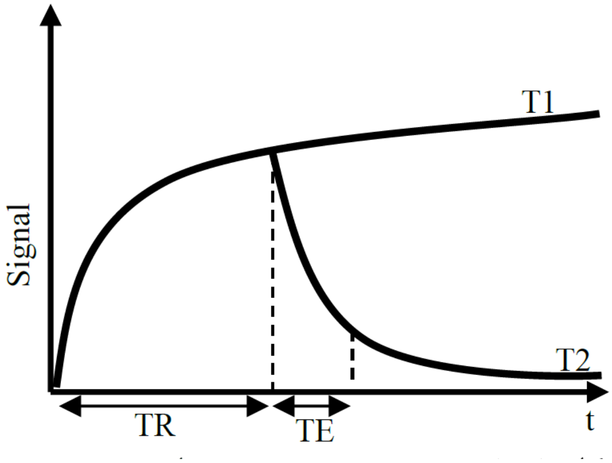


Figure 2-1: TR and TE in MRI Image [21, 22]

Each image slice is coded by the phase and frequency of the received waves on the Y and X axes, respectively. Magnetic fields need to be changed to perform encoding, which is produced at any given moment by superconducting magnets. The information received is entered into a data space called k space and is eventually transformed into image-based instantaneous transformations. Signals received by MRI systems contain a great deal of information that can be used to extract the physical and chemical information of the body or tissue in question. According to the desired information the MRI system is programmed and finally the imaging is presented in one of the following ways or combined methods. MRI images have different sequences, e.g. the one of most common ones is the multi-echo gradient recalled echo (GRE), which is produced by a single RF pulse in conjunction with a gradient reversal. The GRE formation is illustrated schematically [23]. Common techniques used to diagnose MS include T1-W, T2-W, PD-W, and FLAIR[8] images, which are described in the following sections below [24].

### 2-1-1- T1-W

This is a basic method of MRI scanning. For example, the difference between the adipose tissue and the water in this imaging method is that the water is darker than the adipose tissue in the echo gradient imaging with short TE and short TR.

### 2-1-2- T2-W

This method is also one of the main methods of MRI imaging. In this method, like the T1-weighted, the fat and water were separable, with darker fat and lighter water appearing in the image. For example, in studying the brain and spine, the cerebrospinal fluid appears more clearly in the image. Also, in this method, the Eco gradient sequence with long TE and long TR is used. In these images, MS lesions have the highest brightness intensity in the image, because the amount of water in the lesions is greater than the rest of the brain tissue.

---

[8] Fluid-Attenuated Inversion Recovery

### 2-1-3- PD-W

This method is also known as proton density. The contrast in the images obtained by this imaging method is not dependent on the T1 and T2 values and depended only on the density differences of the hydrogen atom which spins in the tissues. Basically, when an MRI sequence is placed to generate a PD-W image, it is the tissues with the higher density of protons (hydrogen atoms) or concentration which produces the strongest signals and is visible as the brightest on the image. In this method TE is short and TR is long.

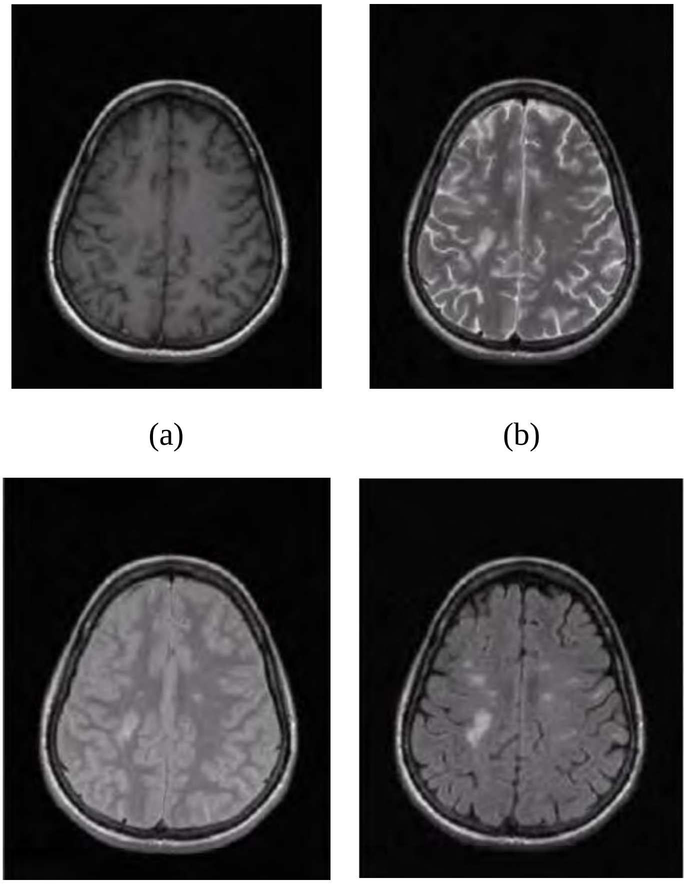
### 2-1-4- FLAIR

This method is capable of displaying small lesions properly. In this image, the signal sent from liquids such as CSF is considered zero with the appropriate TR selection. So, in this image, disturbances close to the CSF[9] are well visible. Figure (2-2) shows an example of PD, T2, T1 and FLAIR images. In Figure (2-2), because of how these images are formed, there is a high contrast between the main brain tissue, the gray area, the white area, and the CSF part of the brain. The CSF area is darker in T1

---

[9] Cerebrospinal Fluid

and FLAIR images, while in T2 images is brighter and, in the PD images the gray intensity is the same.



(a) (b)

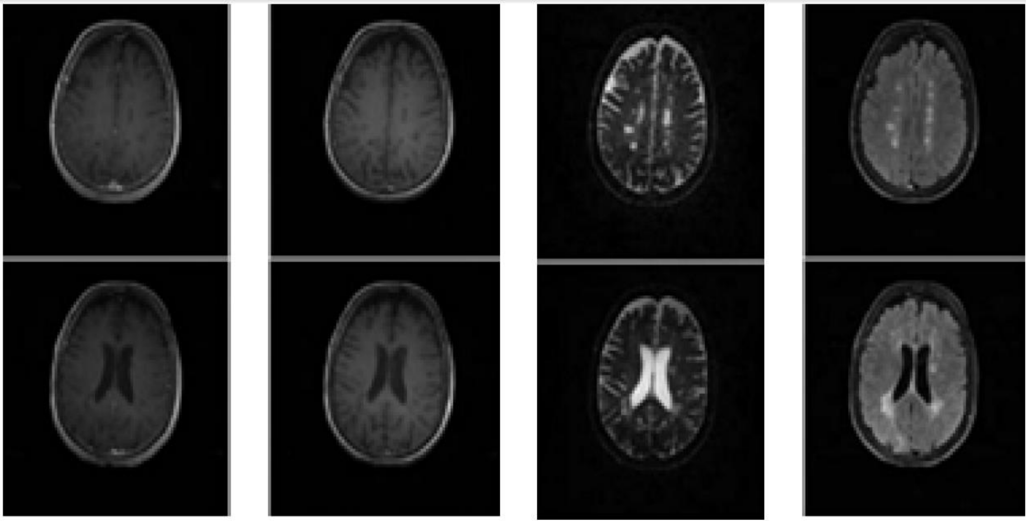
(c) (d)

Figure 2-2: (a) T1-w, (b) T2-w, (c) PD, (d) FLAIR [25]

The white area of the brain in T1 images is the brightest, in PD and T2 images the lowest brightness, and in FLAIR images the brightness is almost the same as in the gray area. Finally, the gray area of the brain is moderately bright in T1 and T2 images. Each image has its advantages and disadvantages, for example T1 images are suitable for tissue anatomy, whereas lesions are more fluid than usual. T2 and FLAIR images have higher brightness and are better detectable in case of lesions. PD images are suitable for both, the study of anatomy as well as lesions.

### 2-1-5- Sequence MRI images

To obtain a patient's MRI images, many sequences of MRI images can be produced. Each MRI sequence provides information on a specific stage of MS lesions. Chronic lesions can be seen in T1-W images and other lesions in FLAIR and T2-W images (Fig. 2-3). High inflammation, edema, and demyelination, which are symptoms of an acute lesion which appears in T1-W images with low signal changes. As a result of these symptoms, T1-W images have the same or higher brightness intensity compared to the white area of the brain. In addition, the boundaries of these lesions are blurred and cannot be traced well. As the demineralization process progresses as well as the chronicity of the disease, the symptoms gradually become darker in T1-W images. Increased demineralization increases the clearness of the lesion boundaries, and this process leads to the formation of some of the chronic symptoms known as black holes. These plates will appear as high-brightness areas in T2-W images [26].



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

Figure 2-3- A sample cut of MRI images with MS lesion a) T1-w, b) T2-w, c) PD, d) FLAIR [26]

## 2-2- Segmentation MRI images for the diagnosis of MS

The task of white matter cells is to carry signals between the gray matter areas and other parts of the body, where they are processed, and the peripheral nervous system seldom engages. These white matter lesions are visible on conventional MRI images. MS shows itself on MRI in the form of numerous rounds, elliptical white spots on the white matter of the brain, so-called plaque. Due to the non-invasive MRI monitoring, this technique has been used extensively for clinical and investigational researches of MS. It is difficult to accurately identify and diagnose MS lesions in MRI images, and segmentation may be error-prone. An accurate alternative to brain segmentation is computer segmentation, which can be done more accurately and in a shorter time for the physician.

In fact, segmentation of an image is the separation of the image into different regions so that the pixels of each region share a specific property (which can belong to an object). The most important feature in segmenting a single image is the color intensity of the image in a color image (the color components of that image). The edges of the image and its texture are also important features in the segmentation [27].

The purpose of segmentation is to break down an image into different fragments and to facilitate its analysis. Image segmentation is usually done to find an object or a series of borders (lines, curves, etc.) more precisely in images. Image segmentation is the process of assigning a label to each pixel of an image, so that pixels with similar labels have similar properties [27].

MRI segmentation means splitting the MRI image into several sections (usually belonging to different tissues). One of the important pre-processors is the processing and utilization of MRI images for various applications such as diagnosis of disease and their process or examination of the impact of treatment.

For several reasons, segmentation of MRI images faces problems that have not been fully resolved, first of all, that data acquisition is always subject to problems. In other words, the taken images are subject to various damaging factors such as noise, non-uniformity of brightness, and so on. This is especially true for MRI images, which essentially use magnetic fields with high non-linear properties.

The second reason is that, tissue segmentation (especially soft tissue) is fundamentally self-operative with uncertainty. There is generally no definite boundary for the regions of each tissue and consequently no definitive answer for their segmentation. This difficulty is not only limited to automated methods. In semi- automated methods, human experts may also have doubts about the segmentation of these images. It may even be possible for an expert to differentiate a picture in different ways.

### 2-2-1- Basic concepts

Before radiological equipment was visualized on film by experts and specialists, the advancement of digital images in radiology departments made it possible to manually segment the lesions and estimate the Total Lesion Load (TLL). However, manual segmentation of the MS lesion was time consuming and laborious. Several researchers have proposed semi-automated segmentation methods to reduce segmentation time and reduce its variability [28]. Ultimately, the main goal is to obtain an automated, repeatable segmentation method that can process hundreds or thousands of images needed for scientific or clinical applications [16].

Due to the heterogeneity of the lesions and their diversity, no satisfactory solution has been found in collecting magnetic resonance (MR) images. Much work has been done to improve segmentation quality, and more recent research on more sophisticated approaches to addressing variability has been proposed. However, validation is still limited and is probably due to the provision of a valid and appropriate framework.

Several comparisons have been made between different methods and only a few are available for free. In 2011, a review article was presented that included some semi-automated and automated segmentation methods [29]. However, in this research, while a broad description of the different methods presented, has a major limitation in which automated and semi-automated methods are mixed.

In fact, segmentation of these images is one of the important ways to diagnose diseases and also to evaluate how tumors grow in medical imaging. Image segmentation involves two steps: first extract the appropriate features from the image and then classify these features using a suitable method. So far, many studies have been done on the choice of classification method. Some of the most important techniques used in segmentation of MRI images are: Thresholding methods (dividing an image into two or more classes of pixels), Area Growth, EM method, Classifiers, Clustering methods, Artificial Neural Networks, and Shape models, and in most cases, image brightness is used as a feature. However, the noise as well as the heterogeneity caused by the radio frequency coils have made the use of image brightness as a feature, not well suited for image area in MRI [25, 30, 31]. Before discussing the methods briefly, it is necessary to discuss the common steps of most methods, including pre-processing and feature selection.

### 2-2-2- Pre-processing step

Conventional MRI techniques, such as T1-W, T2-W, are very sensitive in detecting MS plaques and can provide a quantitative assessment of the extent of the lesion and its activity [32]. As discussed in the preceding section, it is difficult to classify MS in brain MRI images because of varying imaging parameters, intensity overlap, noise, gestures, echoes, blurred edges, normal anatomical changes, and so on [33]. Before any procedure, two important preprocessing steps are needed: first, removing the damaging factors from the image and second, removing non-brain tissues, such as the skull, from the image. What is clear is that MRI imaging itself can also damage the image. However, in terms of image processing, usually to solve all these problems, the light intensities for each lesion are defined as follows:

$$Y = \alpha X + \beta \qquad (2\text{-}1)$$

Where $X$ is the true intensity, $\alpha$ is a smoothly increasing bias string representing noise. $\beta$ is a sequence of a Gaussian distribution. There are various ways to correct $\alpha$, as with many image processing applications, the choice of all pre-processing steps is the balance between algorithmic complexity (e.g. number of adjustable parameters), flexibility, and processing speed.

### 2-2-3- Selection of features in segmentation of MS disease images

Based on the assumption that different brain tissues have different gray surface intensities, common features are used to classify the lesion. In addition, the appearance of the tissues and lesions depend on the image used in MR. (as in Figure (2-1), which shows four samples of different MRI scans of a patient with MS lesions). For example, white matter appears as the brightest texture in T1 images, the darkest texture in T2 images, and the average gray area in the FLAIR image. In contrast, cerebrospinal fluid (CSF) is the brightest tissue in T2 images, whereas in both T1 and FLAIR images is the darkest tissue. Gray matter (GM) appears as a medium gray in both T1 and T2 images, while slightly brighter than WM in the FLAIR image. On the other hand, in MRI images, lesions can appear as light (white) or dark disturbance signals depending on the properties of the lesion and the type of MR image used [25, 34].

# Chapter 3

## Background and Related Work

## 3-1- Literature Survey - Introduction

The main aim of this literature survey is to identify different related issues in Multiple Sclerosis and MRI imaging and after expressing clustering theories, SVM and CLA as algorithms, methods and techniques in MS detection as well as identifying areas for future research. Although in this research, many solutions were identified, it has become apparent that much of the research is being done only relates to the theoretical side of this issue. Thus, this review shows that, however many of the solutions and techniques have been identified, the future research should focus more on the practical implications.

### 3-1-1- Literature Search Method

The first stage is implemented to identify relevant literature. This phase was conducted with the help of provided online database from different publishers, universities, and search engines. The focus in this search was to find the state of the arts regarding Multiple Sclerosis Lesion Detection, Machine Learning Algorithm, Clustering, 3D to 2D MRI Images and Binary Pattern Classification. It is very significant and important to note that review of all the found results in different databases and search engines is impossible from a practical standpoint. It is also important to mention that based on the search way, search engines operate and order the results depend on the computer used for search and many other search settings and factors. (Table 3-1) shows search keywords, online research database and search engines which are used in this literature study.

Table 3-1: Online research resources and keywords

| **Search criterions** | **Name** |
|---|---|
| Publishers | Taylor & Francis – Wiley- Springer – Elsevier |
| Online database(s) | Scopus – ScienceDirect – Web of Science – IEEEXplore – ACM Digital Library – DBLP – MathSciNet – arXiv |
| Search engines | Google Scholar – Academic Search – Microsoft Academic search – WorldWide Sience – CiteSeerX |
| **Keywords relevance to:**<br>Multiple Sclerosis Lesion Detection, CLA, SVM, Clustering<br>Binary Pattern Classification, 3D and 2D MRI | |

Figure 3-1 shows the inclusion and exclusion criterions in order to select related studies for this literature review.

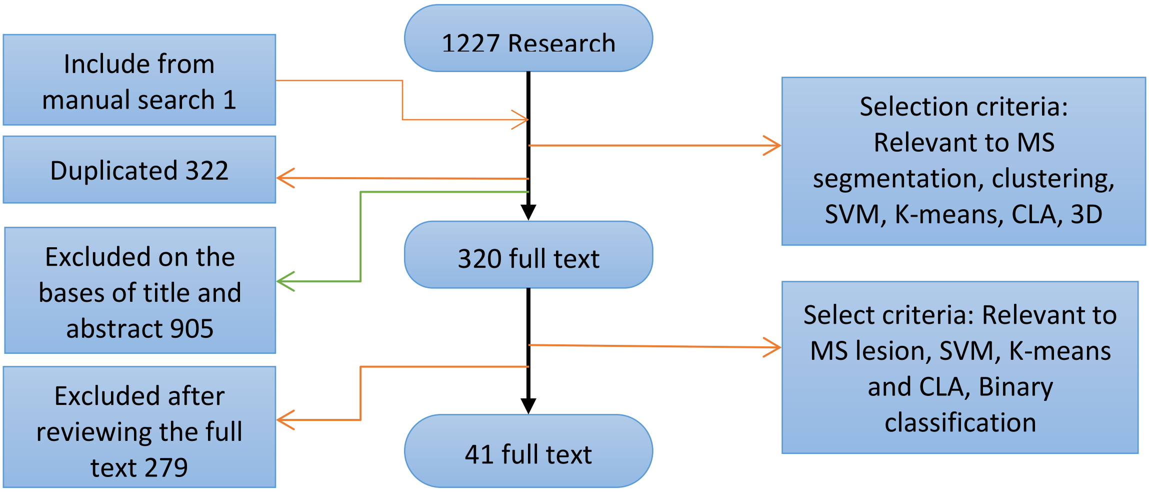


Figure 3-1: The process of selected articles

## 3-2- MRI Imaging Methods for MS Lesions

In this section, we will review some examples of brain MRI imaging segmentation techniques for the diagnosis of MS lesions.

### 3-2-1- Region growing method

In this technique, sub-groups are gathered into larger areas based on a predefined criterion. In fact, initially there is a set of grain points created by adding pixels that are adjacent to the grain and this is based on a predefined criterion that has similar properties to that grain which is creating larger areas. In fact, the region growing technique is one of the best techniques used to classify MRI images. In this way, the interconnected areas of a brain tissue such as a tumor are extracted by a predefined criterion, such as brightness information or brightness intensity and so on. First, a small dot on the image is selected as the seed according to the previous criteria by the operator. The operator's criteria for selecting the seed can be to grow the seed in a particular area of the image until it reaches the edge. Then all the pixels attached to the point that belong to a particular texture are extracted. This method is usually suitable for drawing simple and small structures such as tumors and wounds. This method has many advantages, disadvantages and limitations. The major limitation of this method is the need for the supervisor to accurately select the seed spot, so that each seed that needs to be expanded in the image needs to grow a seed spot. The new sensitivity is to create a cavity or discontinuity in the image [35].

### 3-2-2- Segmentation by integration and division

Another convenient way to segment MRI images is to split and merge. In split and merge, images are split into pieces and then merged to satisfy a set of conditions. The algorithm first duplicates the image into two parts and then merges adjacent areas if conditions are favorable. This process continues until it is no longer possible to divide and merge.

### 3-2-3- Clustering method

Clustering is another practical way to segment MRI images. A cluster is called a set of points or nodes. Clustering allows mapping to multiple servers (cluster points). Clustering is generally a process that organizes members of a larger set into sub-collections with similar properties. Clusters act in many ways similar to segmentation but are of a non-supervised type, this desirable clustering feature is not predefined. This is due to the lack of sequential trained data between the data and the properties that the cluster needs to be. These methods educate themselves through existing data. Clustering has two major parameters. These two parameters are intra-cluster and inter-cluster distance, one of which must be increased and the other to be reduced. In order to perform optimal clustering, the inter-cluster distance should be maximized, and the intra-cluster distance should be reduced to minimum, in which case the best clustering is performed. Examples of conventional clustering methods are the K-Means algorithm, the generalized K-Means, FCM algorithm, and the EM algorithm similar to the FCM method. The number of classes for MRI images of the human brain is set at three because the healthy human brain contains the spinal cord tissue, the white part (white matter) and the gray part (gray matter) [36].

### 3-2-4- EM Segmentation method

Another example of MS lesion segmentation methods is the EM algorithm, in which the lesion is maximized using log likelihood between MRI data (T1, T2, FLAIR) and a Gaussian model of ten classes CSF, GM, WM and six GM / CSF partial classes and one off-bound data class. Figure (3-2) shows an example of WM, GM and CSF masks using this algorithm. In this way, the probability of each voxel belongs to different classes. The EM algorithm repeats two steps after initial processing. At the maximization stage, the parameters of each class (mean and covariance) are calculated from the voxel intensities and their probabilities belonging to different classes. Partial-volume class parameters are calculated as a scale of pure texture parameters, while class parameters are obtained with outliers as part of the CSF parameters; therefore, in the second step, the probability of each voxel belonging to the classes Variables are updated using the Gaussian function with class parameters and previous probability values.

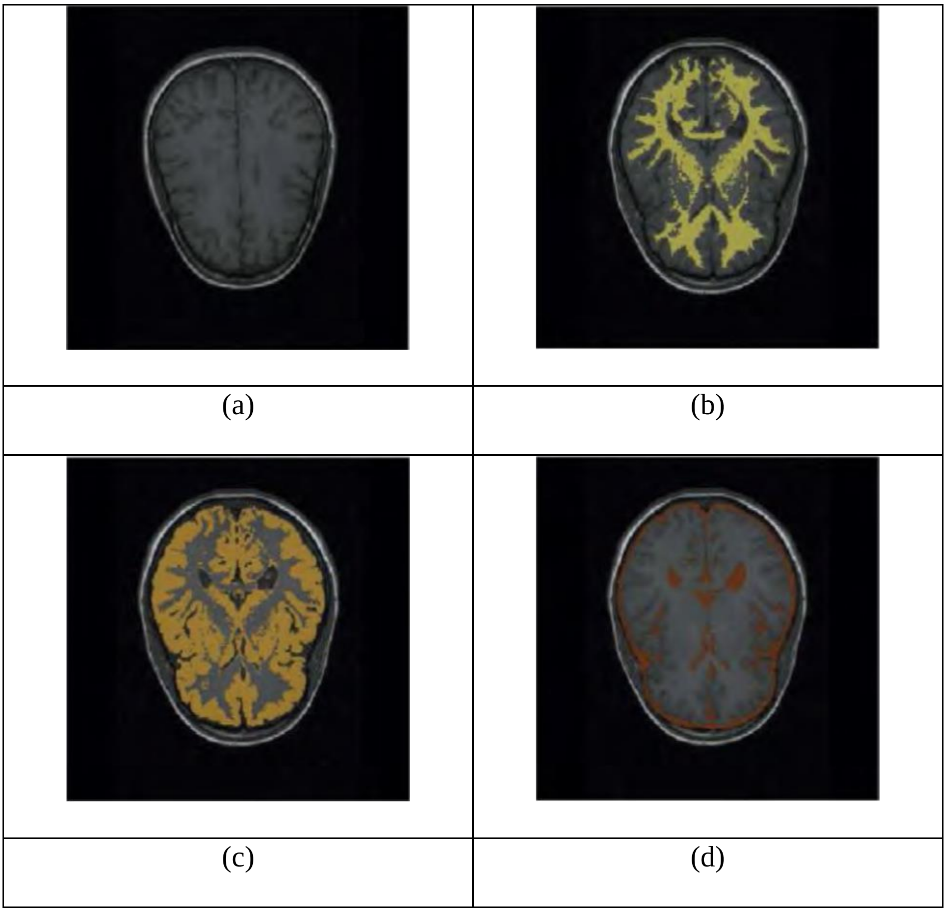

(a) (b) (c) (d)

Figure 3-2: (a) An MRI cut of the brain and symptoms of (b) white matter, (c) gray matter, and (d) cerebrospinal fluid [25]

Finally, when the algorithm converges, a two-dimensional Gaussian distribution for each class including CSF, WM, GM and partial volume values as well as all voxel probabilities associated with that distribution are estimated [25].

### 3-2-5- Histogram (threshold) method

This method performs the zoning using a color histogram. In Fig. (3-3) it is assumed that the image has different brightness levels that shows the peak histories and valleys in the histogram and the boundaries of the tissues are separated at the valley site, and several threshold levels can also be used to select the commonly used method.

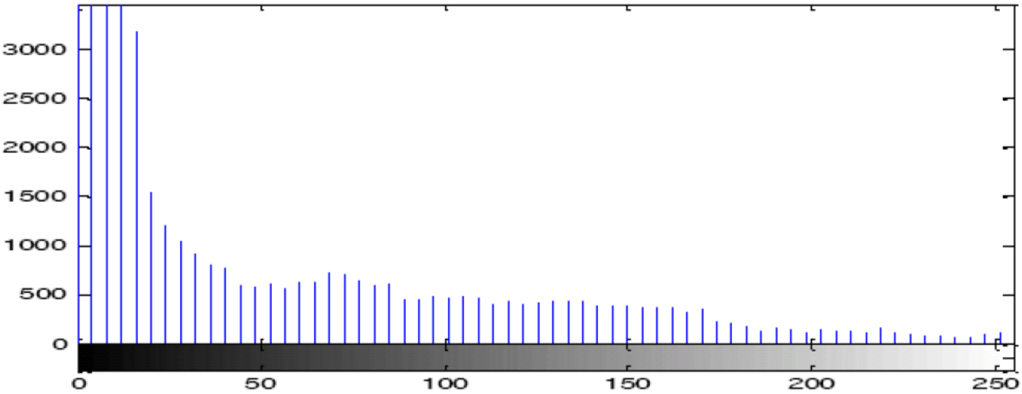


Figure 3-3: Histogram of image brightness [36]

The first stage uses visual processing tasks. In this case, the number of zones considered for zoning is one unit greater than the threshold points. This method is able to segment two healthy and tumor tissues into two distinct classes. In fact, if the image resolution is appropriate, this method is able to clear the subject from the background. This method only uses image histogram brightness information so it does not include image location information [36].

### 3-2-6- Soft Computing method

Three different approaches to soft computing for segmentation of MRI images that are most commonly used for comparison are: 1- Fuzzy-based methods 2- Genetic algorithm-based methods and 3- Neural Network-based methods. The following is a description of these methods [37].

#### 3-2-6-1- Fuzzy-based methods

The various components of fuzzy image processing allow us to develop new image segmentation techniques. Fuzzy approaches to image segmentation are: fuzzy clustering algorithms, fuzzy law-based approaches, fuzzy integrals, fuzzy criteria, image information and fuzzy geometry. The fuzzy set theory provides a mechanism to illustrate ambiguities within an image. Each pixel of an image has a degree of belonging to each of the areas or borders of the image. A number of different fuzzy approaches to image segmentation are described in [38]. Fuzzy entropy is a function on the fuzzy set that decreases when the resolution of the fuzzy set arguments improves. The concept of entropy in fuzzy set theory was first introduced by Luca and Termini. The fuzzy entropy problem has many applications in image segmentation [39]; therefore, the segmentation problem can be formulated as an optimization problem. In fact, fuzzy entropy of an image is chosen as the objective function. One approach in designing such fuzzy systems is for an expert designer to examine training data and thereby to manually create a set of fuzzy rules. Two major disadvantages of such a method are: first, it is very difficult and time consuming and second, there is no guarantee that the produced fuzzy rules are the best possible ones, so to achieve such an approach it is necessary to have fuzzy rules and membership functions automatically created. Various methods have been proposed in the literature for the automatic generation of fuzzy rules, including genetic algorithms and ANFIS [40]. One of the drawbacks of these methods is that there is no guarantee that the fuzzy laws will generate a large number of processing and classification speeds [37].

### 3-2-6-2- Genetic Algorithm-based methods

The genetic algorithm is able to overcome many of the deficiencies found in other optimization techniques, such as comprehensive techniques, calculus-based methods, knowledge-based techniques (law production system, and heuristics) [40]. Because of the generalizability of the genetic algorithm, it is independent of the image segmentation technique and only requires a criterion to measure its performance that relates to the quality of the segmentation. Genetic algorithms are used to solve various problems in machine vision including image segmentation, feature selection, image matching and object recognition. Recently, researchers have used the genetic algorithm to solve the problem of image segmentation. Many pattern recognition applications have used this algorithm. One reason (among many other reasons) for using this approach is the ability of the genetic algorithm to deal with complex search spaces in situations where only minimal knowledge of the objective function is available.

### 3-2-6-3- Neural Network-based methods

Neural networks are composed of several elements with links of different weights. Artificial neural networks have wide applications in pattern recognition. Their potential and nonlinear processing features are used for clustering. Self-Organizing Feature Map (SOFM) is a powerful tool for clustering. Jay and Park proposed an algorithm for image segmentation based on Self-Organizing Map (SOM). This method detects the areas of the image from the brightness component of the color image.

The SOM network is used to solve this image segmentation problem in [37].

## 3-3- Clustering and the Unsupervised Learning

Clustering is an unsupervised learning algorithm and a common technique for data analysis that is widely used in image segmentation, bioinformatics, pattern recognition, statistics, and so on. In fact, data is clustered into meaningful groups or clusters. The contents of each cluster are called clustering features that are similar and yet not identical to other objects in other clusters. Such algorithms are used in large datasets and in cases where there are many data attributes. This method of grouping-similar-documents is based on their content so that documents that have similar properties and features will be grouped together. The purpose of clustering is to place similar documents in one cluster so that they are different from those in the other clusters. The most important algorithms used for clustering are the K-means algorithm. The difference between clustering and classification is that at the beginning of the computation, the clusters are not known and not predefined. In other words, it is not clear in advance the clustering of the groups, and it is also unclear as to which grouping properties [41].

### 3-3-1- Introducing the K-means algorithm

This is a basic method for clustering, which is used in the development of other methods. This algorithm is implemented in different ways but with a consistent procedure, the main purpose of this algorithm is to cluster the data in such a way that there is the least inter-cluster similarity and the highest intra-cluster resemblance. By cluster means the distance between two centers of the cluster and by the cluster means the distance between the center of the cluster and the points assigned to that cluster. Initially, the value of K is given as input and a number of points are randomly selected and an initial clustering is performed based on the proximity of the points to the center of the clusters. The clusters are updated and after updating the points. The center of the clusters is changed using averaging, and the cycle is repeated again. This process will continue until there is no change in taste. The most famous criteria for calculating distance in the clustering method are Euclidean distance and Hamming distance criteria [42].

### 3-3-2- Compare the work of researchers

In general, clustering algorithms are divided into two general categories. The one that runs on a single machine, and the techniques that run on multiple machines that perform better in scalability and responsiveness-time. They provide better users.

In 2013, Iglesias and Kastner, developed the BIRCH algorithm, which has its own data structure and based on clustering. This algorithm has two main steps that first identify the data points and place them in a tree structure of memory and then perform clustering based on the leaves of this tree structure [43].

In 2013, Shirkhorshidi et al., invented an algorithm called CURE, which considers a set of scattered points as a cluster. This algorithm proceeds hierarchically and first uses a point as a cluster and then integrates the different clusters by calculating the distance between the points of the different clusters and finding the minimum distance between the points [44].

In 2013, Silva et al., Researched dimensionality methods, first calculating the correlation matrix from the records in the original dataset, and then extracting eigenvalues for that matrix. These eigenvalues will then be multiplied by the initial matrix and eventually the element space attribute will be mapped to a new space and these attributes will be sorted and based on these attributes the clustering will occur [45].

In 2013, Anchalia et al., developed a non-self-parallel distribution algorithm that, although complex in development, has been greatly improved in terms of speed and efficiency. They used the DBDC algorithm, which is a volume- and distribution-based algorithm. In this algorithm, the distribution is based on the density of points. This method uses compression clusters and is used when there are outliers in the data [46].

In 2013, Barrachina et al., applied techniques to improve speed and scalability; these algorithms consider data samples rather than clustering the data as a whole and then Their clustering generalizes the results across the data, and because they perform clustering algorithms on smaller pieces of data, they increase speed and have less complexity. They used clustering algorithms based on random sampling, commonly referred to as CLARANS. This algorithm reduces the complexity and execution time required for the total number of data objects and is complex [47].

In 2014, Yuwono et al., applied methods to optimize the K-means algorithm to reduce dependencies and thus improve function performance. This change in performance is based on the speed improvement in the algorithm implementation. In this method, steps are implemented following the implementation of the mapping-reduction function to optimally integrate the results [48].

In 2014, Chadha and Kumar, proposed a method that eliminates the dependence on the value of K and does not need to be input to the algorithm. In this case, two clusters are generated initially. Then, it is based on the classical K-means algorithm, and then on each of these clusters, a new data center computation and interval computation is performed so that none of the clusters has a point out [49].

In 2015, Debatty et al, proposed a method to optimize the K-means algorithm for implementation via the reduced mapping function. This method makes changes to the initial selection of cluster centers to improve function performance. In this method, the distance between the sample points is calculated based on the Euclidean formula and stored in a matrix called D. Set A is initialized on the basis of the lowest distances, and set C is considered as the center of the clusters, then the next steps of the algorithm are given by matrix D, and the second lowest distance in each cluster is averaged. The distance between all points is compared, if it is less than the average, is added as the least distance to set A and goes to the third point in terms of the lowest value in matrix D, and if it was greater than the average, then the title of the center of the cluster is added to C and there is no need to recalculate the distances between the sample points. Repeat this operation until the number of C members is equal to the number of clusters or K [50].

In 2017, Dierckens et al., presented a research, data science, and engineering solution for fast data clustering of K-Means, designed in a way called CluDataSE, based on exploratory and hybrid prototypes. Clustering methods are based on spatio-temporal density and are also faster than similar methods [51].

### 3-3-3- Select the appropriate method of clustering

The most well-known clustering algorithm was evaluated, based on the work done, a method was evaluated to improve the initial input clustering by eliminating the dependence on the K-input of the input operation and to appropriately initialize the clustering speed and improved performance and better accountability (Table 3-2). In comparison with the work of the researchers listed in (Table 3-2), we also found that by adjusting the initial clustering, the tolerance to error is also improved. Therefore, the method (Dirkens et al., 2017) is a suitable method.

## 3-4- Supervised learning and classification

SVM[10] is one of the most supervised learning methods used for classification and regression and can be said to be one of the most accurate and robust data mining algorithms. The support vector machine approach is that in the training phase, it tries to select the decision boundary in such a way as to maximize its minimum distance to any of the categories concerned. In this method, it is attempted to implement a system of minimum capacity, or better yet, a system of minimum complexity to obtain class boundaries. As a result, SVM can accurately estimate system boundaries using less training data than competitors, without compromising system generalizability. The support vector machine algorithm falls into the category of pattern recognition algorithms and can be used wherever it is necessary to identify patterns or classify objects in specific classes. In the reference [52] to classify benign or malignant tumors, support vector machine classification is used. In the reference [53] an SVM classification method based on the features, extracted by the dual complex wavelet transform was proposed, but they believed that it would perform better in other approaches. In order to improve the classification

---

[10] Support vector machines

speed in SVM, reference [54], a textured energy criterion law of images is considered.

Table 3-2: Comparison of researchers' work in the field of clustering

| *No.* | *Author/Developer* | *Target* | *Year* | *Simulation tool* |
|---|---|---|---|---|
| 1 | Iglesias and Kastner | Determine the effect of similarity effects in clustering to detect agents when correlations such as factor time series are important. | 2013 | Matlab |
| 2 | Shirkhorshidi et al. | Investigating the Process and Advancement of Clustering Algorithms to Counter Large Data Challenges | 2013 | Matlab |
| 3 | Silva et al. | Work on dimensionality reduction methods in clustering | 2013 | Matlab |
| 4 | Anchalia et al. | Provides a parallel & non-parallel distribution algorithm to improve speed and efficiency | 2013 | Matlab |
| 5 | Barrachina et al. | Designing machine-based techniques to improve speed and scalability | 2013 | Java |
| 6 | Yuwono et al. | Optimization of k-means algorithm to work on time aspects involved in data stream clustering | 2014 | Matlab |
| 7 | Chadha and Kumar | Reduce dependency on K value and design algorithm based on K-Means that does not require number of K clusters as input. | 2014 | Matlab |
| 8 | Debatty et al. | Optimization of the K-Means Algorithm for Implementation through the Mapping Reduction Function | 2015 | Matlab |
| 9 | Dierckens et al. | Algorithmic design to increase speed and reduce runtime | 2017 | Java |

### 3-4-1- Introduction to SVM

A support vector machine is one of the techniques for machine learning to be used for both, categorization and for regression. This method is based on Vladimir Vpnick’s statistical learning theory. The basic idea of this classifier is to find an optimal super-page as a decision-making level in such a way to maximize the margin between the two classes. The data will be mapped to a higher dimensional space with a non-linear kernel if data is not linearly linear, and the optimal super-page will be determined in the new space.

Suppose each data belongs to a group of two positive and negative groups. $L$ can be called educational data represented by the regularized pairs $(x_i, y_i)$, so that $x_i$ is the vector of the n-dimensional property and $y_i \in (-1, +1)$ is its label. The goal is to find the hyperplane that separates the two classes with the -1 and +1 labels with the largest margin. This super-page can be expressed by the equation.

$$w^T\varphi(x) + b = 0 \tag{3-1}$$

In this regard, the vector $w$ is the vector perpendicular to the hyperplane, b is the bias vector used to measure the distance from the hyperplane to the source and the kernel, $\varphi(x)$ is for mapping the data to a higher-dimensional space (Fig. 3-4).

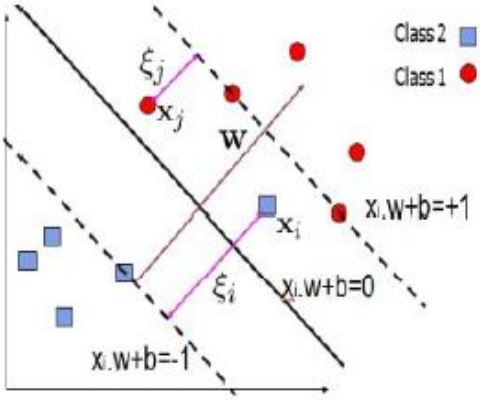


Figure 3-4: Maximizing the margin between two classes is equivalent to minimizing $w$, which leads to solving the minimization problem.

$$\min \frac{1}{2} w^2 + C \sum_{i=1}^{N} \xi_i \qquad (3\text{-}2)$$

$$\text{Subject to:} \ y_i(w.\varphi(x_i) + b) \geq 1 - \xi_i, \forall x_i, \xi_i \geq 0, y_i \in \{-1, +1\}$$

That parameter C is called the parameter in the backup vector machine. The $\xi_i \geq 0$ variable is used to consider the noise in the data and the interference between educational data. The existence of a condition guarantees that the data is not in the margin. In support vector machine, tags will be given as inputs to the problem. What makes clustering differentiate the problem is that in the clustering problem, the differentiation of the vector of the tags will be also considered as unknown to the problem.

$$W_{nm}^{i} = \exp(-\lambda_p \|p_n - p_m\|_2^2 \; - \lambda_c \|c_n \; - c_m\|_2^2\,) \tag{3-3}$$

Accordingly, the problem of optimizing a normal cut is given in the following way to the similarity matrix, where L as a normal Laplacian matrix will be:

$$\begin{gathered}\min_{\mathrm{y}} \frac{\mu}{\mathrm{N}} \mathrm{tr}\left(\mathrm{y}^{\mathrm{T}}\mathrm{Ly}\right)\\ \mathrm{L} = \mathrm{I} - \mathrm{D}^{-\frac{1}{2}}\mathrm{W}\mathrm{D}^{\frac{1}{2}}\end{gathered} \tag{3-4}$$

A differentiation clustering method seeks to solve an optimization problem in which data is tagged in a way that produces the least error. In the following, equation (3-5) represents a distinct unit cluster. Subsequently, the combination of this relationship and the equation (3-4) to the segmentation algorithm will be defined.

$$\begin{gathered}E_U(y) = \min_{\substack{w \in R^{K\times d}\\ b \in R^{K}}} \frac{1}{N} \sum_{n=1}^{N} l\left(y_n, A\varphi(x_n)\right) + \frac{\lambda}{2K} \|w\|_F^2\\ E_U(y) = tr(y^T A y)\\ A = \lambda\left(I_N - \frac{1}{N} 1_n 1_n^T\right)(N\lambda I_N + K)^{-1}\left(I_N - \frac{1}{N} 1_N 1_N^T\right)\end{gathered} \tag{3-5}$$

Under the circumstances, the combination of this relationship and the relation (3-4) to the segmentation algorithm will be defined. The optimization problem limit must be defined in this manner as to make ideal clusters in connection with the number of Data. The optimal clustering relationship is defined as the distinction for the segmentation problem as follows:

$$\min_{y\epsilon\{-1,1\}^n} y^T \left(\mathrm{A} + \frac{\mu}{\mathrm{n}}\mathrm{L}\right) y,$$
$$\text{Subject to:} \, \forall \mathrm{i}, \lambda_0 \mathrm{n_i} \delta_\mathrm{i} \leq \frac{1}{2}\left(\mathrm{yy^T} + 1_\mathrm{n} 1_\mathrm{n}^\mathrm{T}\right)\delta_\mathrm{i} \leq \lambda_1 \mathrm{n_i} \delta_\mathrm{i} \tag{3-6}$$

As we can see in the optimization question, the vector y defined as the vector of the tags is considered to be the problem unknown.

### 3-4-2- The most important types of support vector machines

#### 3-4-2-1- SMO[11]

Sequential Minimal Optimization (SMO) is widely used to train support machines and has been implemented by the very popular LIBSVM package[12]. The publication of SMOs in 1998 caused a great deal of public acceptance by the SVM community[13], as existing methods for solving SVMs were costly and complex [55]. As a matter of fact, SMO is a simple algorithm that quickly solves the QP[14] SVM problem without any matrix storage and without the use of numerical solution and quadratic programming optimization. The SMO selects the smallest problem in each

---

[11] Sequential Minimal Optimization

[12] Library for Support Vector Machines

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

[14] Quadratic programming problem

step for optimization. For the standard QP SVM problem, the smallest possible optimization problem involves two Lagrange multipliers, since the Lagrange coefficients must follow the equilibrium linear constraints. At each step, the SMO selects two Lagrangian multipliers for optimization. SMO continually detects two Lagrange coefficients that can be optimized with reference to each other and analytically calculates the optimal stage for these two. Hence, the original QP problem is solved, when there are no two Lagrange multipliers to be optimized. The advantage of the SMO method is that the solution of the problem for the two Lagrange multipliers can be obtained analytically. Therefore, throughout the solution QP optimization is numerically avoided. Moreover, as mentioned, this method does not require memory to store the matrix, so large training sets can also be managed by this method. The amount of space required for SMOs is linear relative to the size of the training set, which allows the SMO to manage large training sets. As this method avoids matrix computation, the SMO falls between linear and second order relative to the size of the training set, while in standard SVM, it will fall between the second and third order. In the real world, SMOs can be up to 5 times faster than the standard segmentation algorithm for SVM [56].

### 3-4-2-2- CVM[15]

Vladimir Vapnik in 2013 introduced the Core Vector Machine (CVM) method as a fast way to train SVM with very high-volume data. The standard SVM has the training time of the $Q(m^3)$ time order and the spatial complexity order $Q(m^2)$ (m is the size of the training set). The CVM method can be applied to nonlinear kernels and has linear temporal complexity of $m$ and spatial complexity independent of $m$. Experimental results have shown that in large amounts of data, CVM accuracy is as good as SVM, while being able to handle large amounts of data at higher speeds. The algorithm presented by CVM to optimize and find the best approximate solution to the optimization problem is implemented by obtaining a set of vectors called Core Vector. This algorithm, despite its simplicity, guarantees an approximate solution, also has short time and minimal spatial complexity [57].

### 3-4-2-3- Reduced SVM (RSVM)

The motivation for this approach is to solve large data set management problems with SVM including non-linear Kernel. This method selects a subset of data as backup vectors and solves the problem

---

[15] Core Vector Machine

of smaller optimization. Experimental results show that RSVM accuracy is slightly lower than standard SVM but can be useful in problems with tens of thousands of data volumes. Faced with a large body of data, RSVM has been proposed as a solution to reduce computational complexity and reduce model complexity [58].

### 3-4-2-4- RVM[16]

Relevance Vector Machine (RVM) is a machine learning approach that offers low-cost solutions to probabilistic regression and classification using Bayesian inference. RVM has a functional form similar to SVM but provides probabilistic classification [59].

### 3-4-2-5- TWSVM[17]

TWSVM is a convenient machine learning approach and suitable for classification and regression problems. The purpose of TWSVM is to create two asymmetric pages for each class by optimizing a pair of QPPs[18]. In this way, each hyperlink is close to the data samples of one class and away from the data of the other class [60]. Thus, TWSVM solves a pair of smaller sized QPPs instead of a complex QPPs in conventional SVM.

---

[16] Relevance Vector Machine

[17] Twin Support Vector Machine

[18] quadratic programming problems (QPPs)

### 3-4-2-6- TBSVM[19]

According to the machine, the dual boundary vector support is linear and non-linear. This is an improved TWSVM approach and is a significant feature of TBSVM over TWSVM in minimizing structural risk. Experimental results related to this method show that the computation time is low and the classification accuracy is high [61].

### 3-4-2-7- LSTSVM[20]

In LSTSVM or at least squares of the support vector machine, quadratic programming is used to solve the machine model. When we calculate a large-scale problem, we may find it costly. In this regard, the least squares method was designed for support vector machines. LSTSVM has less computational complexity, higher accuracy and higher generalizability than TWSVM but still has problems with over-resolution and noise sensitivity. Subsequently, however, improved LSTSVM was introduced, which reduced the problem of over-resolution. Following this approach, the LSTSVM was tuned to perform better than the TWSVM and eliminate the sensitivity to the noise effect but still has difficulty depending on the choice of weight parameters. Finally, multi-class

---

[19] Twin Bounded Support Vector Machine

[20] Least Squares Twin Support Vector Machine

LSTSVM, which has lower computational complexity and higher classification accuracy, was presented which has the problem of non-classifying parts [62].

### 3-4-2-8- v-TWSVM

While TWSVM reduces the experimental risks associated with training data patterns, v-TWSVM removes this weakness of TWSVM by setting two parameters for margin error control, as well as reducing the number of backup vectors. The generalizability of v-TWSVM is better than conventional TWSVM [63].

### 3-4-2-9- DTTWSVM[21]

DTTWSVM is used for multi-class problem (K>2). In the decision tree, higher nodes have a higher degree of efficiency in classification. There are two things to creating a binary tree: A) two groups of base classes have the best resolution, when the two groups have a maximum distance between all classes. B) Two groups of classes have the best separation rule when these two groups have the lowest non-categorization rate among all classes [60].

---

[21] Decision tree based twin support vector machine

### 3-4-2-10- MLTSVM[22]

A multi-label learning paradigm, which seeks to segment data that potentially contains several categories. In 2016, Chen et al. Presented MLTSVM for multi-label classification. MLTSVM defines multiple non-parametric cloud to obtain multi-tag information embedded in the data. Experimental results on real and fictitious multi-label datasets confirm that this method is applicable and highly efficient. Learning in multi-label data has also created challenges for traditional non-virtual cloud compartments. In this method, we assume that each sample belongs to a particular category only, and the decision-making process is based on the distance of the samples to the cloud and overcomes the ambiguity of the test procedure [64].

After comparing the work of the researchers listed (Table 3-3), assessment is made of support vector machine algorithms, which according to the table results, MLTSVM Multi-Label Learning Model (Chen et al., 2016) is the best choice.

---

[22] A novel twin support vector machine to multi-label learning

### 3-4-3- SVM and MRI for MS Detection

The brain MRI image processing using artificial intelligence (AI), were the topic of numerous research papers. In Performance Analysis of Machine Learning Classifiers for Brain Tumor MR Images investigated cancerous and non-cancerous brain MRI images using ANN, SVM, K-nearest Neighbor (KNN), Decision Tree (DT) and Naïve Bayes (NB) classifiers and compared the effectiveness of these methods. In this research the final conclusion was that ANN reached the best accuracy which is up to 87.4%, in case of binary brain tumor identification for low grade and high-grade tumors. However, it must be noted that this accuracy depends on the size of the dataset [65, 66]. At the Department of Computer Science of the University of Cyprus a **dataset** is available containing MS MRI Lesion Segmentation. The dataset for related research was published all in the topics of texture analysis and classification. The conclusion of these papers that MRI is indeed a powerful tool in MS diagnosis and lesions which differentiate mild and severe MS can be separated using texture analysis and classification methods [67-70]. The article by Abdullah, Younis, Pattany and Saraf-Lavi, (2012 pg. 26) provides comprehensive and detailed perspectives about the textural based Support-Vector Machine (SVM) [71].

Table 3-3: Comparison for support vector machine algorithms

| No. | Algorithm/technique/model | Target | Year | Simulation tool |
|---|---|---|---|---|
| 1 | SMO (Sequential Minimal Optimization) | In each iteration two variables are selected, and the problem is solved analytically. The amount of space required is linear relative to the size of the training set. In the management of large training complexes, SMO is faster than SVM. | 1998 | Matlab |
| 2 | RVM (Relevance Vector Machine) | Maintains high performance and privacy of SVM RVM using Bayesian inference offers low cost solutions for probabilistic regression and classification. | 2001 | Matlab |
| 3 | RSVM (Reduced SVM) | In problems with large data volumes or with a large number of backup vectors, RSVM can be useful as a solution to reduce computational complexity and reduce model complexity. | 2007 | Matlab |
| 4 | v-TWSVM | Greater accuracy, reduced computational complexity, better generalization performance, smaller support vector than TWSVM | 2012 | Java |
| 6 | DTTWSVM (Decision tree based TWSVM) | Better generalizability compared to other multi-class categorization methods, low computational complexity, short learning time | 2013 | Matlab |
| 7 | TWSVM (Twin Support Vector Machine) | It's faster than traditional SVM. It is more accurate. It's relatively easy to teach. Good scale for high dimensional data. | 2015 | Matlab |
| 8 | TBSVM (Twin Bounded Support Vector Machines) | Faster than TWSVM, better generalizability, higher accuracy | 2015 | Matlab |
| 9 | MLSTSVM | The lower the computational complexity, the more accurate the classification. | 2016 | Matlab |
| 10 | MLTSVM | Multiple non-parallel super-Page definitions can be implemented and very effective | 2016 | Matlab |

The purpose of using the support vector machine is to ensure the all the elements undertaken in the health institution are digitized. Therefore, the application of this digitization technique is the fact that it provides the users the chance to examine all the spinal features by making use of the digital aspect rather than requiring manual conformation on whether the entire process has been success. In addition, the digitization technique happens to provide comprehensive and analytical information concerning the situation of the region of interest (ROI) [72].

Another aspect which can be perceived from the implementation of the system is the fact that it contains a post processing module, a feature which adequately works to improve the segmentation quality. Specifically, the article by Abdullah et al., (2011 pg. 26) provides a comprehensive overview of the effectiveness of making use of his imaging technique in address MS cases. The article makes use of primary data where 10 sets of MRI images were used in training of the trained support vector machine (SVM) [71, 73]. The fascinating aspect of the study is the fact that the overall results showcased 0.68 for the level of sensitivity, 0.79 represented the level for die similarity whereas 0.9 represented the overall percentage of detected load of lesion. Therefore, from the statistics and explanations it is correct to note that the stated data image processing technique can be applied in digital processing of MS cases [71]. The figures in that paper showcases the lesion segmentation process. However, the statistics showcase that there is adequate room for improvement that can be undertaken in the research. Therefore, one can conclude that the textural based aspect cannot be treated as the most viable technique that can be applied in this case.

It is evident that the incorporation of computer aided systems is paramount in this generation and thus, the same can be extended into the medical facilities. The other comprehensive digital processing technique is the K-Nearest Neighbor (K-NN). The K-NN can be described as one of the less sophisticated classification techniques as it is used in the evolution of the required data based on the voting function and the distance function [74, 75]. Specifically, the digital pattern tends to make use of a certain statistical pattern so as to provide detailed information about the existing MS condition. The statement can be evidenced by the fact that the image classification technique happens to make use of a statistical recognition method where there is a set of samples with already existing figures in a pre-existing manner. Therefore, the health practitioner collecting sample of the image from the patient with the MS condition happens to compare the raw data with the existing sample which happens to contain K samples of learning. Thus, the practitioner needs to have sound and articulate knowledge of the samples and thus, this provides the individual to determine the condition of the patient by determining the closest sample considering the Euclidean distance.

Another technique which has been proposed is the Ada-Boost Algorithm which combines the statistics collected from other weak classifiers and uses them to come up with a comprehensive and diagnostic assertion concerning the MS condition in an individual. According to D’cruz and KV (2015 pg.2) [76], the technique makes use of the linear combination of the inputs from other weaker classifiers. The primary concept enshrined in the imaging technique is that once the information from the weaker classifiers has been merged the measure of central tendency can be established and thus, this leads to the formation of a strong imaging technique such as the SVM. Therefore, the examples of weak classifications include the Bayesian networks and the decision trees. Despite the fact, these procedures can be perceived as an effective imaging technique, in the real sense, its application can be perceived as misplaced. For instance, the decision trees are fundamental elements in the decision-making process but one of the major demerits associated with such procedures is the fact that there is high level of inaccuracy associated with such techniques [77]. In this case, the combination of the results from the weak classifiers cannot be perceived as an appropriate technique especially while aiming to treat individuals diagnosed with multiple sclerosis [78]. Therefore, this shows that there is need to come up with other comprehensive imaging techniques which can provide the health practitioner an effective way of handling cases of multiple sclerosis.

## 3-5- Cellular Learning Automata (CLA)

Cellular Automata (CA) consists of a regular network of cells, and each cell has a limited number of modes, such as on and off, unlike a paved network. The network can have a limited number of dimensions. For each cell, a set of cells is defined as the neighbor. In the initial mode (time t = 0), a state is assigned to each cell. A new generation (t of 1) is created by some rules (in general, a mathematical function) when the new state of each cell is determined based on the current state of the cell and the status of the neighboring cells. Cell status updates are the same for each cell and will not change over time and apply to the entire network simultaneously [79]. CLA can be considered as a single object with finite number of operations. The learning automaton operates by selecting an action from its set of actions and applying it to the environment. The action is evaluated by a random environment and automatically uses the environment response to select its next action. During this process, automata learns to choose the optimal operation. How to use the environment response to the automata selective action used to select the next automata action is determined by the automata learning algorithm [80].

CA cannot solve many of the collective problems. By combining the two models of CA and LA, a new model called Cellular Learning Automata (CLA) is made. The aforementioned mathematical model is very powerful for solving many decentralized problems. CLA is a CA in which one or more learning Automata are assigned to each cell. At the same time, the CLA must operate under specific rules. These rules are set by the neighboring learning atom using a booster signal in each cell. In addition, in accordance with these rules and the preceding states of the neighbor, the learning atom in each active cell receives a reward or a fine. Granting a fine or reward updates the structure of the CLA [81]. In $d$ dimension CA, we have the structure of $CLA = (Z^d, \phi, A, N, F)$ where, $Z^d$ is a network of d-tuples of an integer that can include a final network, a network be infinite or semi-finite. $\phi = (1, \dots, m)$ is a finite set of states. A is a set of LA, each assigned to a CLA, each cell having at least one LA or more.

$N = (x_1, x_2, \dots, x_n)$ is a subset of $Z^d$ and is known as the neighboring vector ($x_i \in Z^d$), and F is the local law of CLA.

### 3-5-1- Clustering according to the Learning Automata

Clustering can be considered as the most important method in unsupervised learning. In clustering, the concern is to divide the data into clusters so that the similarity among the data of each cluster is at its maximum, and the similarity within the clusters is at its minimum. Clustering follows an iterative procedure that tries to estimate the followings for a considering number of constant clusters: obtaining some points as the centers of the clusters, that in fact these points are the means of the points belonging to each cluster. Assigning each sample data to a cluster where the data was closest to the center of the cluster [82, 83]. In the simplified type of this method, first, depending on the number of required clusters, some points are randomly chosen. Then, the data are assigned to one of the clusters according to their nearness (similarity), and thus new clusters are formed. By repeating this procedure, in every iteration, through obtaining the mean of the data, new centers can be calculated for the data, and they can be assigned to the new clusters again. This procedure would go on until no change would take place among the data [84-87].

CLAs are a mathematical model for the systems that are dynamic, complex and consist of very simple components, which have learning capability, and behavior of each component is determined and corrected by the behavior of its neighbors and its past experiences. The simple components which form this model can show a complex and powerful behavior through interacting with one another [83, 86, 88].

Pixel based methods judge whether a pixel is a lesion pixel only by its color, while region based methods combines color and texture information [89]. Since using neighborhood and texture information can be practical in the skin detection, and since CLA can easily use neighborhood information [83], it can therefore be a suitable candidate to be employed for lesion detection. CLA is composed of the components that are in the same neighborhood and the behavior of these components is defined according to the current behavior of their neighbors and their own and their neighbors' experiences (reward and punishment that they receive for their actions and reinforcement signals that they receive from environment).

CLA is a hybrid model of two, learning automata and cellular automata. In the following these two models are discussed. Cellular automata are a mathematical model that can be used for the simulation calculations of the systems. Cellular automata are simple discrete systems that can exhibit complex calculations and behavior through simple and local rules. Locality in the cellular automata means that in determining the new value of each cell, the cells in its neighborhood are used and more distant cells would have no usage. Each cell has a set of states for itself and at every moment, according to its own state and the state of the neighbors, it decides to what state it goes. State switch rules are constant for cellular automata and never change. Cellular network can have different dimensions and be one, two or multi-dimensional [89]. According to the values that cells can possess, cellular automata are classified to two types of binary and multi-valued. Understanding the behavior of the cellular automata from its rules is very difficult and requires simulation. One of the problems of using cellular automata is designing rules that perform our desired action [88]. There are different types of cell updating rules that cause different types of cellular automata to be generated. For example, the rules can be expressed in probabilistic and deterministic forms and these two types of rules lead to two types of deterministic cellular automata and probabilistic cellular automata. Characteristics of the cellular automata can be briefly expressed as follows: automata are homogeneous and updating would be performed synchronously. Of course, asynchronous updating would also be possible, i.e. we start updating from one corner of the network and continue to the end, while in the synchronizing state, all the cells become updated according to the previous state of the neighbors not according to their present state. Dominant rules are defined according to the neighbors of each cell and can have deterministic or probabilistic form. Learning automata are applied in many cases, such as physical processes simulations like Brownian motion, dissolution, social processes simulation like rumor dissemination, chemical phenomena simulation like fire spread and iron corrosion, image processing, random number generation, and cryptography [82, 85]. Given its rules, learning automata can show complicated behaviors. Rules in the learning automata can be defined as a bit string in which each bit represents the next state corresponding to the number of the bit [90, 91].

Van Leemput et al. proposed a method of classifying intensity-based tissues using a random model for normal brain image. This procedure simultaneously identifies MS lesions as outliers that are not well described by the model. He first diagnoses them as Outlier according to a statistical model for MRI images. He then uses the brain atlas, which contains information about the location needed for a variety of tissues. This method corrects the heterogeneity of MRI images and classifies the textual information by the hidden Markov model. This is done on images T1_weighted, T1_weighted, PD_weighted. The advantage of using it is high correlation but has less spatial correspondence than the method recognized by the experts [92].

De Boer et al. used PD_weighted and T1_weighted images for segmentation of CSF, WM, GM. Initially, the images were pre-processed with non-uniform intensity correction and the images were recorded with high and low illumination intensity. On average, they created a Tissue probability Maps (TPMs) for all four tags (background, GM, WM, and CSF). These TPMs were threshold for candidate training samples, with a predefined probability of belonging to a specific label. In this method a threshold of 0.7 is selected. A tree spanning a minimum of specimens is created in the feature space to remove duplicate connections. Finally, all unbound samples or those that are not in their original cluster are deleted. It is then applied to the modified set of K-nearest neighbors (KNN) classifiers. A histogram of voxels was created in images classified as GM. This histogram is smoothed by Gaussian function and the peaks in the histogram are defined as the region with the highest Voxel. This method yields good segmentation results and has the advantage of being independent of a set of training. This method has the potential to be easily applied to imaging data with new scanners or parameter settings of the new scanner [93].

Abdullah et al. proposed a new technique for the automatic segmentation of multiple sclerosis (MS) lesions on magnetic resonance imaging (MRI). This is done with a trained Support Vector Machine (SVM). The distinction is made between MS lesions and non-MS lesions on the basis of tissue characteristics and other features. Support Vector Machine (SVM) is a supervised learning algorithm that at its core generates a method for generating a predictive function from a set of training data. The radial basis function (RBF) is selected as the core. The main feature of the proposed method in this paper is the use of textural properties. The tissue characteristics used distinguish between white matter, lesions, natural white matter and gray matter. Texture classification has also been used for analysis. Shear classification was performed independently of each section of brain MRI. The texture extraction method is based on an optimized filter designed to identify the ROIS. MS lesion detection is a fully automated approach. The texture-based SVM method proposed using three datasets of MS, Synthetic Data, MS Subjects MRI Lesion Segmentation Challenge 2008., Data Acquired at the University of Miami, simulated. Performance evaluation and comparison of results with other automated techniques shows that this method provides competitive results for the diagnosis of MS lesions [71].

Datta et al., propose a way to obtain images at higher resolution because low resolution images result in misclassification. In the first step, the brain is classified into GM, WM, CSF, and the lesion using non-parametric techniques (Parzen classification). This method has poor performance in people with low-volume lesions [94].

Roura et al. developed an algorithm consisting of two main steps. The first stage is brain tissue segmentation according to gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF) in T1_weighted images and the second stage is lesions as outliers in the GM region in FLAIR images. Brain tissue segmentation is achieved using T1_weighted images. It is then used to find the GM distribution in FLAIR images and to obtain the lesion mask as a hyper-intense outlier vector. The method was investigated on three databases: 5-T dataset, 3-T dataset, MICCAI2008. This method achieves the best results in ranking using an unsupervised strategy. This simple duplication strategy enhances performance, especially on the 3-T dataset and challenges images with artifacts and abnormal intensity [95].

Brosch et al., used a neural network approach consisting of two interconnected paths. A convolution path abstractly describes high-level image features and a deconvolution path that predicts segmentation at the voxel level. Feature Extraction and Route Forecasting Tutorials are used to automatically learn features at different scales for each combination of image types. In addition, the shortcut key between these two paths allows for high and low-level features that are capable of segmenting lesions. This method is applied to two datasets MICCI2 2008, ISBI2015. The results show that this method is higher than the other methods. Even when a relatively small dataset is available for training. In addition, the combination of nonlinear processing units is strong enough to learn features that, under large variations, are crucial for the segmentation of MS lesions. This method is performed with the network, at different depths with and without the shortcut key, and the results show that increasing the depth from three to seven layers improves performance and increases accuracy by adding the shortcut connection [96].

## 3-6- Evaluation

Performance evaluation requires two stages of evaluation, including segmentation performance evaluation and classification evaluation. To evaluate the performance of segmentation and analyze the results of the segmentation method, the criteria of overlap and similarity of two binary images of the relations 3-7 and 3-8 have been used.

$$\mathrm{DICE(A,B)} = \frac{2|\mathrm{A} \cap \mathrm{B}|}{|\mathrm{A}| + |\mathrm{B}|} \tag{3-7}$$

$$Jaccard = \frac{|A \cap B|}{|A \cup B|} \tag{3-8}$$

A is the image of the fragmentation (real image) and B is the image of the output of the new fragmentation technique. To evaluate the classification performance of sensitivity, accuracy, Specificity and classification error, the relations 3-9 to 3-12 have been used.

$$\mathrm{Sensitivity} = \frac{TP}{(\mathrm{TP} + \mathrm{FN})} \tag{3-9}$$

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{3-10}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3-11}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(O_i - F_i)^2 \tag{3-12}$$

TP[23] Correct identification of MS lesion as MS, TN[24] correctly identifies normal brain as normal, FP[25] Misdiagnosis of normal brain as MS lesion, FN[26] Misdiagnosis of MS lesion as normal brain, O is the target and F is the output matrix [97].

## 3-7- Overview and Comparative Study

MRI imaging is able to display MS lesions at different intensities than its surrounding tissues. Careful assessment of any MS lesion in MR images manually for medical professionals is a laborious task and a mentally divisive division. An attractive and accurate alternative to manual segmentation is computer segmentation. This can be done more accurately and in less time for the practitioner.

In their research, researchers have focused on improving the classification of brain lesions, especially those from MS, which is a relatively difficult task, using different algorithms, but in determining the edge. They have problems, they use to solve the problem and divide based on the edges, but in different areas it is possible to ignore and not label.

---

[23] true positive
[24] true negative
[25] false positive
[26] false negative

The gap in previous research is the lack of using advanced learning methods, so according to the research background, we have collected methods for using various tools of learning methods, which have already been used to diagnose MS lesions. In this study, we want to test other combinations of these learning methods. To do and develop this, we start with simple learning methods such as k-means to find MS lesions, and then combine the different learning methods step by step to reach a suitable result in the accuracy of the diagnosis. Therefore, a combination of clustering, SVM and CLA methods is suggested which will be discussed in Chapter 4.

# Chapter 4

## Proposed Method

## 4-1- Methodology - Introduction

In this chapter, after expressing clustering theories, SVM and CLA, different algorithms combining these methods are proposed to find an automatic and learning method for the automatic detection of MS lesions and at the end of this chapter, different MS lesion detection scenarios are outlined.

## 4-2- Unsupervised learning system and K-means method

The most important and widely used algorithm in clustering is the K-means algorithm, which is one of the partitioning methods where each cluster is represented by its mean objects (cluster center). This algorithm works well when the clusters are separated into compact clouds. This method is effective for relatively large databases, but often results in a local optimization. One of the disadvantages of the method is to determine the number of clusters that must be known in advance and no efficient method has been provided. It is also not suitable for detecting clusters with complex shapes. Another major disadvantage of this approach is its sensitivity to remote data. These data easily change centers and may not produce the desired results [97].

In the K-means algorithm, *K* first randomly selects a member (where *K* is the number of clusters) and considers them as the centers of the clusters. The remaining *N* - *K* members are then allocated to the nearest cluster. After the allocation of all members, the cluster centers are recalculated, and the members are allocated to the clusters according to the new centers, which will continue until the cluster centers remain constant with no change. In other words, if we have data sets of *n* data points $(x_1, \ldots x_n)$ and number of *k* clusters $(C_1, \ldots, C_k), k \leq n$, the K-means goal is to minimize the mean squared of the similarity interval $\|x_j - c_i\|^2$ [98];

$$Min \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - c_i\|^2 \qquad (4\text{-}1)$$

In Eq. (4-1), to find the centers of the *K* cluster, the Euclidean mean squared distance must be minimized, between a data point and the nearest cluster center. The K-means algorithm provides an easy way to implement an approximate solution of the equation. The reasons for the utility of the K-means method are its ease, simplicity, scalability, convergence speed and consistency with sparse data. The K-means clustering algorithm can be considered as a slope or descending gradient method. Cluster centers begin and regularly update cluster centers in order to reduce the objective function in the equation K-means always converges to the minimum location.

The location of the minimum depends on the starting point in the center of the cluster. The problem of finding the least universal is the problem of completing the NP. The K-means algorithm updates the cluster centers until the minimum location is found. The generalized pseudo-codes of the K-means algorithm are shown in Figure (4-1). The exact value of $L$ varies depending on the starting state of the center of the cluster even in the same dataset. Thus, the computational time complexity of the $O$ ($N$) algorithm, where $n$ is the total number of objects in the dataset, $K$ is the required number of clusters we have identified and $L$ is the number of iterations $(k \leq n, l \leq n)$ [99]. Implementation of the standard K-means involves continuous and sequential repetition. Each iteration of the entire data set visit is required to assign the data to the corresponding clusters. At the end of each iteration the new centers are calculated so that the next iteration uses the new centers. After a certain number of such duplicates the centers will remain the same [100]. The operation of the algorithm must be such that the boundary between the data set that is likely to be switched to another cluster and the data that holds and maintains the cluster that it owns during the next iteration is tracked.

```
Step 1: Accept the number of clusters in the group data and then accept the data set in the cluster as input values
Step 2: Initialize Clusters k
Take k samples or random sampling k
Step 3: Calculate the average figures of each cluster arranged in the dataset
Step 4: K-means assigns each record in the dataset to only one of the initial clusters, each record is assigned to the nearest cluster using a distance measurement (e.g. Euclidean distance).
Step 5: K-means assigns each record in the dataset repeatedly and repeatedly to the most similar cluster and again calculates the mean of all the clusters in the dataset
```

Figure 4-1: Common code in K-means

As the implementation of the K-means algorithm progresses, the centers get closer to their final position. As the number of iterations increases, the centers become less divergent from their current position and consequently fewer data items are controlled and reviewed [101].

According to the algorithm presented in Fig. (4-1), it is obvious that in the case of data set, step 2 requires the highest time frame in the implementation of the algorithm. The larger the data set, the more time frame it takes to execute step 2 because it visits each data subject and performs calculations on it. The question here is: Do we need to visit the entire data space?

In reality, most clustered data subjects or clusters whose centers move slowly should not be affected by motion. They will remain part of the same cluster at the next iteration and have fewer points of motion. The ability to detect which data topics affect motion means that we do not need to visit the entire dataset and only a small part of the dataset is sufficient. Before deciding which data issues to include in the "demarcation", we need to determine the criteria that must be implemented by the data elements in a way that is capable of incorporating the "demarcation" element. Assume point $p$ is part of cluster $C$ (Figure 4-2). Point $p$ is part of cluster $C$ and the distance between $p$ and $C$ boundary is less than its distance to $A$ and also less than its distance to $B$. We want to know how far point $p$ is to jump to another cluster. Clearly, the formula would be:

$$ep = Min(d_{PA} - d_{PC}, d_{PB} - d_{PC}) \tag{4-2}$$

Where $ep$ is the distance between $p$ to the nearest boundary. It can be said that point $p$ is about the size of $ep$ for switching to another cluster. At the end of duplicate $i$, the centers need to be updated and updated according to the new order.

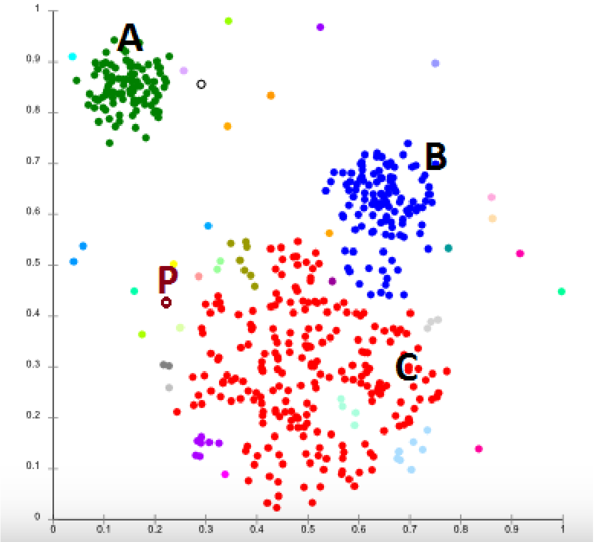


Figure 4-2: Three hypothetical clusters of data

Suppose center $A$ has moved $A$ and center $B$ has moved $B$ and $C$ has moved $C$. The worst-case scenario for point $p$ is that boundary of point $C$ is further away from point $p$ as it gets closer to point $A$ and $B$, so what would be the conditions for $P$ to get closer to cluster $C$, conspicuously it will be as follows:

$$ep > |CC| + |AA| \tag{4-3}$$

$$ep > |CC| + |BB| \tag{4-4}$$

To simplify the algorithm and reduce the computation we can combine terms (4-3) and (4-4):

$$ep > 2 * Max(|AA|, |BB|, |CC|) \tag{4-5}$$

Which is generally a way to determine whether a point is part of a "demarcation" list or not [102]. Because check-and-control returns the differences and inequalities for the data elements in each iteration where we started. To avoid such computations, we can trace all of our data elements at wider intervals for the value as shown in simplified Pseudo Algorithm and block diagram (Figure 4-3). (Figure 4-4), as follows [51]:

```
Define constant WIDTH
Define intervals $li = li * WIDTH, (i+1) * WIDTH$ and tag them with value $i * WIDTH$
Mark the entire data set to be visited
For each point to be visited
Compute $e = min (dpci - dpcw)$ where Cw is the center of the winner (closest) cluster and Ci, $i=1..k$, $l \neq w$ stands for all other centroids
Map all points with $i * WIDTH < e < (i+1) * WIDTH$ to interval $i * WIDTH$ where i is a positive integer
Compute new centroids Cj, where $j=1..k$ and their maximum deviation $D = max(|CjCj'|)$
Update ij's tag by subtracting $2 * D$ (points owned by this interval got closer to the edge by $2 * D$)
Pick up all points inside intervals whose tag is less or equal to 0, and go to 4 to revisit them
```

Figure 4-3: Simplified Pseudo Algorithm

Algorithm group points with values so that we can execute it for the whole group instead of visiting each data element and controlling it against the distance near the edge and their boundary. The key to all optimization is in compatibility performance. Wide compatibility has a great impact on optimization.

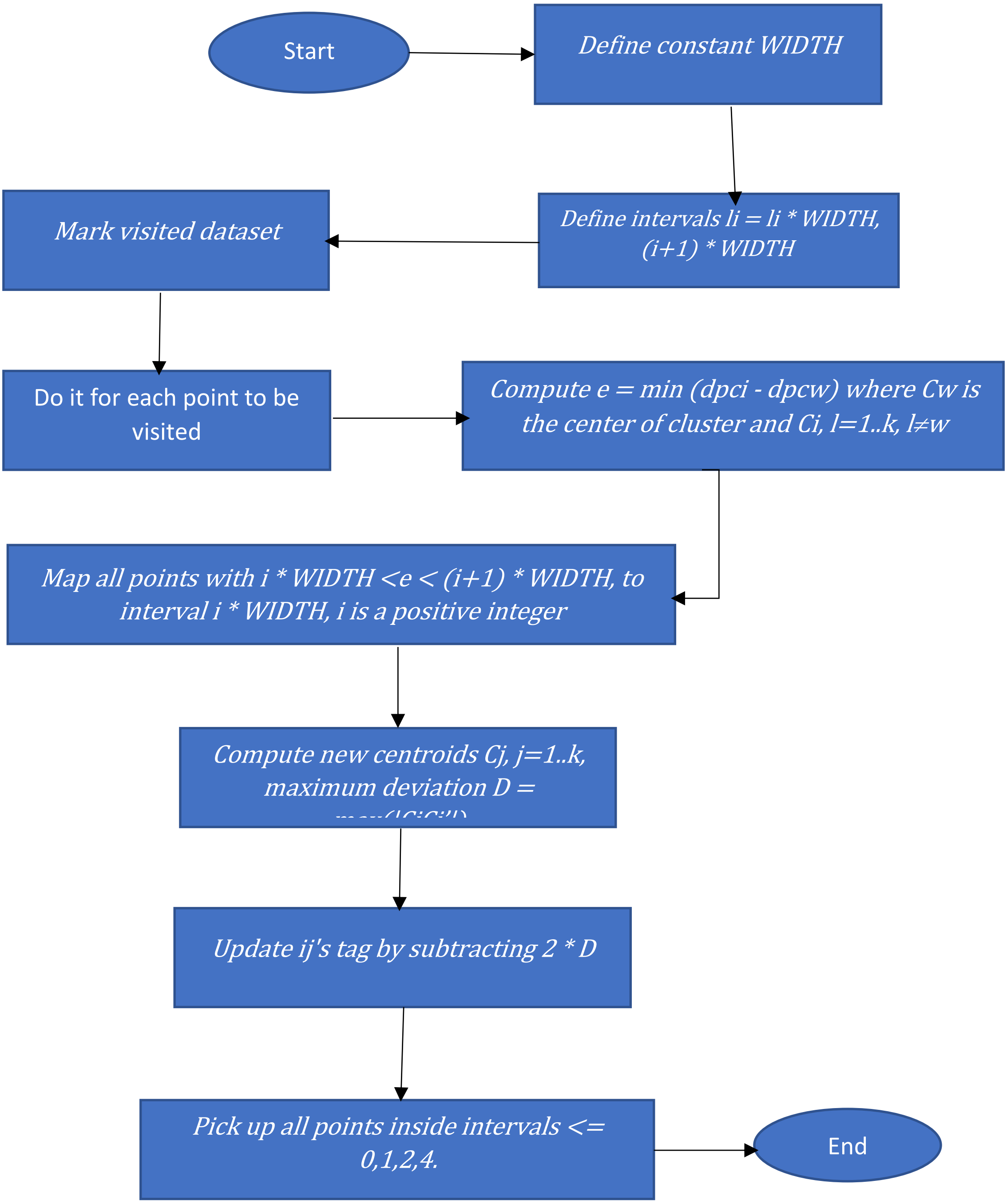


Figure 4-4: Simplified Pseudo Algorithm block diagram [111]

If the value of the width is low, then the number of intervals increases and therefore workload for checking and controlling and updating the intervals in each iteration can be significantly increased. If the value is too wide, the number of points increases for each distance, and although the number of intervals decreases. The absence and loss of execution is obvious when the long distances are re-marked. It is easily noticeable and clear to see that the quality of the final clustering will not be affected by the proposed optimization. In each iteration, clustering and sorting are exactly the same as if standard k-means were used [103].

## 4-3- Determination of K-means Algorithm Starting Points by Cellular Learning Automata

The choice is made by neighboring learning automata. Each learning automata generates a boost signal vector that the other learning automata determine each other's boost signal vector. In cellular learning automata, learning automata are neighbors (a combination of learning automata). Each learning automata forms its own environment, which is a constant around the environment, since the neighborhood probability vector is different [88]. The operation of the cellular learning automata is that in the first step, the internal state of each cell is determined, and the internal state of each cell is determined based on the probability vector of the static learning automata in each cell. The initial value may be randomly selected or by past experience. In the second step, the rule of cellular learning automata determines the amplification signal of the resident learning automata in the cell, so each learning automata is updated on the basis of the amplification signal it receives, and the probability of its action is updated. This process continues until the desired result is achieved [104]. The orders are as follows:

A) Determine the structure of the cellular learning automata: In the first step, the dimensions of the cellular learning automata are determined. The number of elements of one cellular learning automata is the number of records. Each record is assigned a number. We then specify the number of clusters and randomly select points for the number of clusters. The number of elements is the number of clusters. If the number of clusters is $k$, the cellular learning automata is calculated from (4-6).

$$\sum_{i=1}^{n} CLA_i = k \tag{4-6}$$

In the above relation, $n$ means the number of records.

B) Definition of Neighborhood in cellular learning automata: The proposed neighborhood definition is that, every point with all points has a neighborhood. Because we can move the values of a CLA cell to its zero point.

C) End of cellular learning automata: At each stage of the cellular learning automata, the transfer science is performed. After each transfer step, the CLA is evaluated according to the cellular learning automata and the best clustering accuracy is stored. If the transfer occurs 100 times and the best accuracy of the cluster is not changed, the cell learning automata stops (the number 100 is calculated by trial and error). Genetic algorithm as input receives cellular learning automata and with genetic algorithm determines the starting points of K-means algorithm. The best chromosome with output accuracy is selected, then the clusters are plotted, and the average of the clusters is selected as the starting point of the K-means algorithm. Then the K-means algorithm is implemented (Figure 4-5).

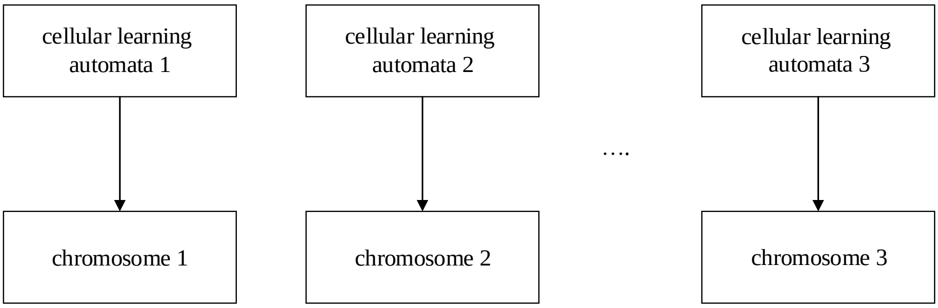


Figure 4-5: Formation of chromosome structure by cellular learning automaton

## 4-4- Proposed method: CLA with k-means clustering integration

CLA is defined as a unidirectional graph in which each vertex represents a cell equipped with a LA. The CLA rule and the LA-selected actions of the neighbor of a particular LA, amplify the LA signal residing in a cell define. The neighboring LAs of each specific LA constitute the local environment of that cell. The local environment is a non-stationary cell because the probability vectors of neighboring LA action change during CLA evolution [104, 105]. The Automata Behavior Assessment Environment is P-Model. In the P-model environment, each action in each iteration receives only a penalty or reward. If LA chooses an action such as $a_i$ at n-th stage and receives a favorable response from the environment, the probability of $a_i$ operation being $p_i(n)$ increases and the probability of other operations decrease. Conversely, if the response from the environment is undesirable, the probability of $a_i$ operation decreases and the likelihood of other automated operations increasing. However, changes are made so that the sum of $p_i(n)$ is always constant and equal to one. In equations (4-7) and (4-8) the update of $p_i(n)$ operation $a_i$ in n-th step is based on the fines and rewards received by the linear model learning algorithm-P. Based on the favorable or unfavorable response of the environment, the equation (4-7) or equation (4-8) is used to update the probability of automata being applied [88, 105].

$$\begin{aligned} p_i(n+1) &= p_i(n) + a\big(1 - p_i(n)\big) \\ p_j(n+1) &= (1-a)p_j(n), \forall_j, j \neq i \end{aligned} \tag{4-7}$$

$$\begin{aligned} p_i(n+1) &= (1-a)p_i(n) \\ p_j(n+1) &= \frac{b}{r-1}(1-b)p_j(n), \forall_j, j \neq i \end{aligned} \tag{4-8}$$

In equations (4-7) and (4-8), $r$ is the number of acts, and the reward parameter and $b$ the penalty parameter. When $a$ and $b$ are equal, the algorithm is called $L_{RP}$, when $b$ is smaller than $a$, the algorithm is called $L_{R\varepsilon P}$ and when $b$ equals zero, the algorithm is called $L_R$. In this study, the number of Automata actions is set to the number of clusters. The reward and penalty parameters are set to $L_{RP}$ algorithm with a value of 0.1. Euclidean distance is used to determine the neighboring cells and the number of neighbors of each cell $i$ is $N_i = 4$. That is, at the beginning of each run, the cell's $N_i$ Cellular Learning Automata is considered the neighbor of the cell based on the lowest Euclidean distance. The steps of the proposed algorithm are as follows:

1. Each data is stored on one cell and each cell is equipped with an LA. The cells are then randomly distributed in space.

2. The probability value of all the functions in Automata of each cell is equal to $1/|K|$, Is set, where k is the number of clusters as $\sum_{i=1}^{n} CLA_i = k$.
3. For each cell $i$, the neighboring $N_i$ with the least Euclidean distance are selected as neighbors of that cell.
4. For a specified number of iterations (ITER) selects the following Automata actions according to one of the following:
    - If ITER=1, it selects a random operation for each Automata in each cell.
    - If ITER>1, it checks if the number of clusters obtained by CLA is less than the number of clusters specified at startup, then selects one operation for each Automata in each cell.
    - If ITER>1 and the number of clusters is equal to the number specified by CLA, the action is most likely to be the next action in each Automata.
    - The environmental response is influenced by the action chosen by the neighbors of a cell. If the greater percentage of neighbors agree on the current cell's selective action, meaning they have chosen a similar action, then the selected action will

be rewarded, otherwise it will be penalized. This is stated in the following law:

- If the action chosen is equal to the action chosen by $n$ or more than $n$, $n = round(cN_i)$ of the neighbors of the current cell, then $\beta_i(n) = 0$ otherwise $\beta_i(n) = 1$, constant and some It has a range of [0, 1], which refers to the neighborhood effect coefficient and is usually greater than 0.5.
- Automata in each cell updates its probability based on the response of the environment and the relationships (4-7) and (4-8).

5. Repeat continues until the environment response to the selected operation of all Automata is optimal. After the loop stops, the most likely action in the Automata of each cell is selected as the data label.

## 4-5- Image processing with CLA

In one image, some pixels in the image have certain properties that distinguish them from their neighbors. These pixels are called feature pixels. The purpose of feature extraction methods is to find and distinguish these pixels from other pixels in the image. Detection of a pixel as a feature pixel is done by cellular learning automata by applying local law to the neighboring space of each pixel. The image is first mapped to a cellular learning automata so that each pixel is assigned to one of the cells of the learning automaton. Each cell's automaton has two actions. One of the actions is the existence of the search attribute and the other is the absence of the search attribute in that pixel. Each cell automatically selects one of its actions and compares it with the action of its neighbors and stabilizes or modifies its action. The neighborhood space of any cellular learning automata in a pixel such as *p* is a circular space at the center of *p* and radius *k* [106].

The local law used to reward or penalize is defined as the number of pixels that have a gray area near the central pixel. If the number exceeds a threshold number, the selected action will be rewarded or fined otherwise. The threshold number depends on the type of attribute desired. If a pixel belongs to the boundary of an area, its neighbors will have a different gray area than that pixel. Gray slope gradient is a good criterion for edge detection. Edge detection gradients measure the gradient and direction of the gray level adjacent to each pixel [107].

Each automata has two actions. Assume the first action equals one pixel to the edge and the second act equals one pixel to the edge pixels. At first, each automata randomly selects one of its possible actions. At startup, we consider the number of cars that choose the first action less than the ones that choose the second. At each stage of repetition, each automata compares its situation with that of its neighbors and corrects its behavior on the basis of this comparison. Here's how each automaton is evaluated and decided at each iteration step:

If a cell in the cellular learning automata chooses its first action, or in other words, recognizes its corresponding pixel as the edge, and if the number of adjacent octaves with the cell that chose the same action is between two to four, then selected actions are appropriate and rewarded. In other words, a pixel is an edge if it has an elbow between two to four neighbors. But if a cell in the cellular learning automata chooses its second action or, in other words, does not recognize its corresponding pixel as the edge, and if the number of adjacent automata's in the eight cells of that cell have chosen the same action, or more than four, the action selected is appropriate and will be rewarded. Otherwise, the selected action will be incorrect and will be penalized. We will repeat the above actions until a certain number of cells reach steady state or in some cases no automatic change.

To detect horizontal lines of learning automata's in each cell, a cellular learning automata considers a pixel to be a horizontal line if the two automata, located on its right and left sides are on that horizontal line. In other words, the gray area difference of a cell with the right and left cells is very low and with many other neighboring cells. Also, the vertical difference of the gray area of a cell with high and low cells should be very low and with other neighboring cells very high. To distinguish oblique lines or oblique edges in the image the difference between the gray areas of a cell with its diameter, cells must be very low and with that of other neighboring cells [108].

## 4-6- Different Scenarios for the Diagnosis of MS Lesions using MRI Images

### 4-6-1- Semi-automatic method and use of K-Means

The method used in this section is the combination of Region growing method and lesion clustering. Figure (4-6), shows the flowchart and the job description step by step.

### 4-6-2- Automatic MS Segmentation Approach Based on Cellular Learning Automata

In recent years, MS plaque detection based on color has attracted the attention of the many scholars for its low computational complexity. Most of these methods have a high error rate for not considering the texture of the lesion. In this paper, a novel method to detect MS lesions is proposed. This method combines the image information in the gray area and texture, detects the skin area by using Cellular Learning Automata (CLA) [83, 85, 89]. Cellular Learning Automata (CLA), the combination of Cellular Automata (CA) and Learning Automata (LA), is preferable to both CA and LA individually [82], [109]. Unlike CA or LA, CLA tries to learn optimal actions and it can improve the learning capability by using a set of learning automata that interact with each other.

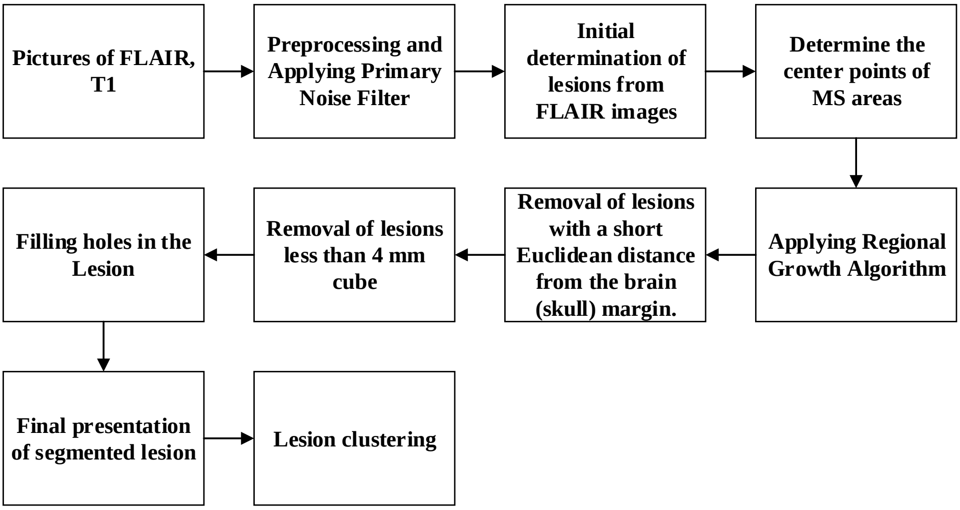


Figure 4-6: algorithm of combination of region growing method and lesion clustering

In addition, CLA can optimize outside standard sequential processes mapping real world solutions more accurately [90, 91, 109]. Firstly, the candidate lesion regions are detected through defining a threshold, and then these regions are assigned to a texture analyzing system. The output of texture analyzing system is the probability rate of being lesion for each pixel. Then this probable mapping is given to a CLA. By using neighborhood relations and texture information, CLA converges to a steady state representing the final output of the proposed system. The proposed algorithm is shown in Fig. (4-7), proposed Pseudo algorithm is shown in Fig. (4-8) and Flowchart of the proposed algorithm in Fig. (4-9) [111]:

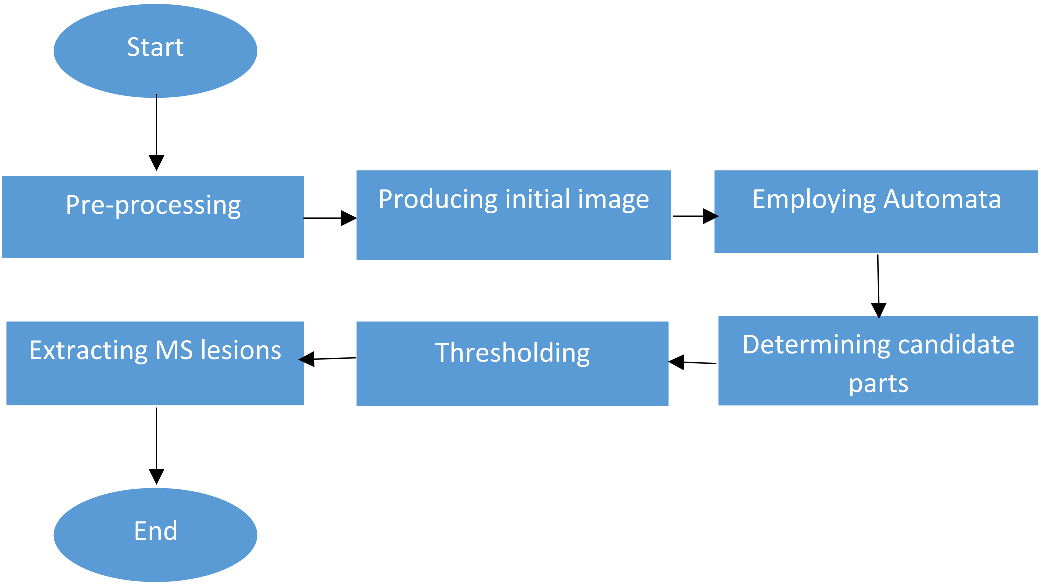


Figure 4-7: Block diagram illustrating the proposed CLA algorithm [111]

```
Function (plaque detection function using cellular learning automata)
Establish an associative CLA.
Initialize the state of cells in CLA.
for each cell j in the CLA do
Let x be a data sample from the data set, give x to cell j
Let i be the class of data x
Cell j selects an action $\alpha j$
 if Cell j is in row i then
   if $\alpha j = 1$ AND half or more neighbors of cell j selects action 1 then
    Reward the selected action of LA in the cell j
     else
      Penalize the selected action of LA in the cell j
   end if
     else
   if $\alpha j = -1$ AND half or more neighbors of cell j selects action -1 then
  Penalize the selected action of LA in the cell j
     else
   Reward the selected action of LA in the cell j
   end if
 end if
end for
```

Figure 4-8- The Pseudo algorithm of the proposed CLA algorithm [111]

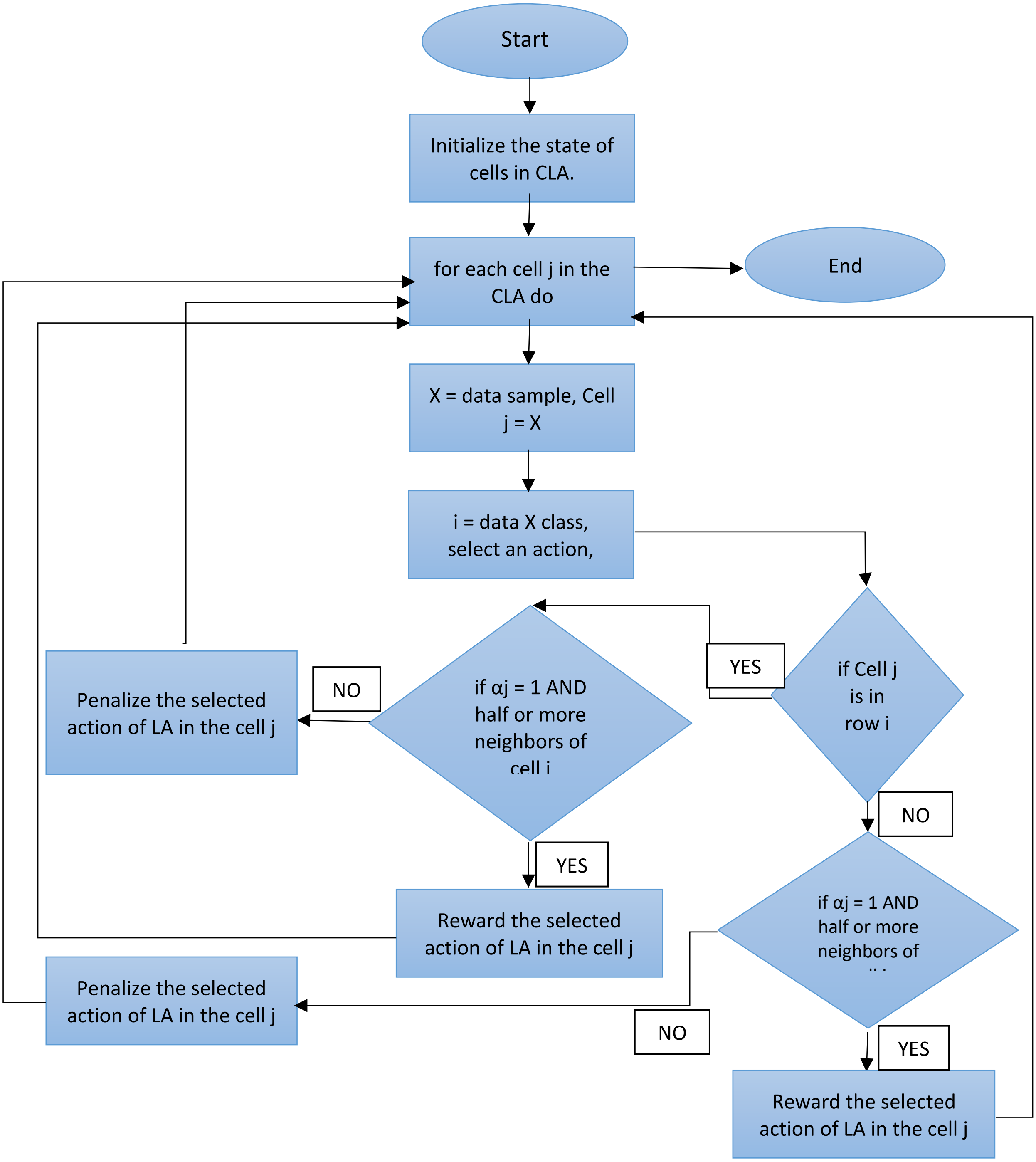


Figure 4-9- Block diagram of the proposed algorithm

### 4-6-3- MS Segmentation Approach based on SVM and CLA integration

Gheshlaghi et al., Presented an approach to improve the detection of MS lesions in brain MRI images using a hemispheric lesion detection and fuzzy algorithm. This paper presents a method to improve the results of MS lesion segmentation system in MRI images. This method consists of pre-processing, feature extraction, classification and post-processing. The skull preprocessor has been removed to improve system speed and performance. In the next step, it is diagnosed to reduce the computational volume of hemispheres with lesions. This is done by comparing the histogram of each hemisphere with the standard histogram that is made up of healthy hemisphere images. Before extracting attributes, each brain image is populated by a slider and the attributes are extracted in each window. The feature extraction section uses Gabor method (Gabor Filter). In order to increase the accuracy of the system, the mean fuzzy algorithm *c* is used in the post processing part. The SVM supervised method was used to accurately evaluate the results of the feature extraction method [110].

Moghadasi et al., the thesis author, showed that the segmentation of the lesions was investigated using gray scale properties and dimensions of the lesions. The researchers used contrast-enhanced MRI images of

T1_weighted, T1_weighted, PD_weighted containing MS lesions. To diagnose MS lesions, the pixels of MRI images were analyzed using the Cellular Learning Automata (CLA) method. MS lesions showed white and red anatomy of the brain on a black background [111].

An automated algorithm for the diagnosis and segmentation of lesions is proposed to increase the accuracy of segmentation of MS lesions. The algorithm consists of pre-processing parts, detecting MS-hemispheres, feature extraction, classification and post-processing. In the pre-processing section, the brightness intensity of the normalized images and the brain region are first extracted. Then, to reduce the computational volume, the lesions are diagnosed. In the feature extraction section, a slider window is used, and the properties are extracted using the Gabor method (Gabor Filter). In the classification section, the window is labeled as a healthy area or MS and the image of the lesion is obtained. The result in the post processing section is improved using the Cellular Learning Automata (CLA) algorithm with k-means clustering (Fig. 4-10).

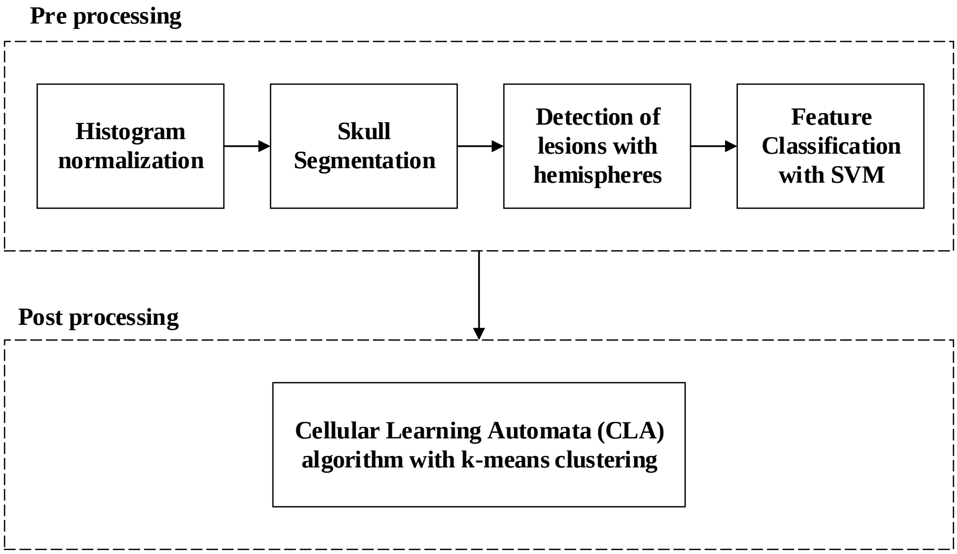


Figure 4-10: Proposed algorithm CLA and SVM integration

In the Controlled High-Risk Subjects Avonex Multiple Sclerosis Prevention Study (CHAMPS) showed, that patients who received early treatment with IM IFNβ-1a developed clinically definite MS (CDMS) with 44% lower probability than in patients who received placebo and changes in lesion volume were significantly different between the two groups. In addition, after 18 months, there were 58% fewer new or enlarging lesions and 71% fewer Gd+ lesions in patients who received treatment than in those who received placebo. A parallel study called Early Treatment of Multiple Sclerosis (ETOMS) also confirmed the finding of CHAMPS. ETOMS concentrated on the group of patients who had uni-focal or multi-focal neurologic syndromes with certain type of lesions.

ETOMS showed that after 2 years of treatment, 24% fewer patients who received treatment developed CDMS than those who received placebo and annual relapse rates were also lessened in the active treatment compared with the placebo group (Miller, 2019) [112]. Before the use of MRI in the clinical diagnoses, patients had to wait for years until receiving MS diagnosis. As it was emphasized, early treatment has a significant role, therefore earliest possible recognition of MS has a great importance (Miller, 2019). Machine learning algorithms proved to be successful in the past in brain MRI image analyzation. This thesis is the continuation of a research by the author who aimed to create artificial models using support vector machines (SVM) to classify MS and normal brain MRI images, analyze the effectiveness of these models and their potential to use them in Multiple Sclerosis (MS) diagnosis. In the published papers, a case study presented at Infocommunications Journal, we intend to show that 3D images can be converted into 2D and by considering machine learning techniques and SVM tools. The first research work concluded that SVM is a potential method which can be involved during MS diagnosis, however, in order to confirm this statement more research and other potentially effective methods should be included in the research and need to be tested. As it is discussed in the previous chapters and literature review and methodology section, this study continues the research of SVM used for classification and Cellular Learning Automata (CLA), then it expands the research to other method such as Artificial Neural Networks (ANN) and k-Nearest Neighbor (k-NN) and at that point compares the results of these, and finally propose combined algorithm for better MS lesion diagnosis. In this published study two approaches were used since SVM is a binary classifier, however the current dataset can be divided into four mutually exclusive classes. In order to resolve this, in the previous study One-Against-One (1A1) and One-Against-All (1AA) techniques were introduced [113]. The goal of 1AA technique to divide the N class dataset into N two-class cases, while 1A1 approach creates a model for each pair of classes so N(N-1)/2 models are built. In this study each method had an equal vote (Moghadasi et al., 2020) [114].

# Chapter 5

## Results and Discussion

## 5-1- Results & Discussion; Introduction

According to the algorithms presented in Chapter Three, this chapter presents the results, and the algorithms are as follows: Semi-automatic method and use of K-Means, Automatic MS Segmentation Approach Based on Cellular Learning Automata, MS Segmentation Approach based on SVM and CLA, Simulated Database 3D MRI to 2D Images, using value of Binary Pattern Classification for MS Detection.

## 5-2- Results for Semi-automatic method and use of K-Means

### 5-2-1- Apply Primary Thresholding and Determination of Primary Lesions

It can be said that it is based on the analysis and analysis of the histogram of the intensity values. The image is made up of luminous objects in a dark background, the gray scale histogram has two maxima. A peak is produced by luminous objects and a courier by field points. If the contrast between objects and context is high enough, the two maxima in the histogram of the image are completely separate from each other, and the threshold intensity is two maxima. In the original image all values greater than Abba show all values less gray and equal to 0. This produces a binary image where the points of the objects are represented by a value of 1. If the image is made up of two components, the histogram has a maximum of 2 and the segmentation is performed with a few thresholds. That is about 140 to 150 in our implementation plan. Table (5-1), shows the photo parameters in the FLAIR channel and Fig. (5-1) is shown FLAIR images and their initialized thresholding.

Table 5-1: FLAIR image parameters in channel in used tools

| | |
|---|---|
| Repetition Time | 9000 ms |
| Echo Time | 103 ms |
| Inversion Time | 2500 |
| Matrix Size | 256×204 |
| Echo Train Length | 17 |
| Imaging Frequency | 123 |

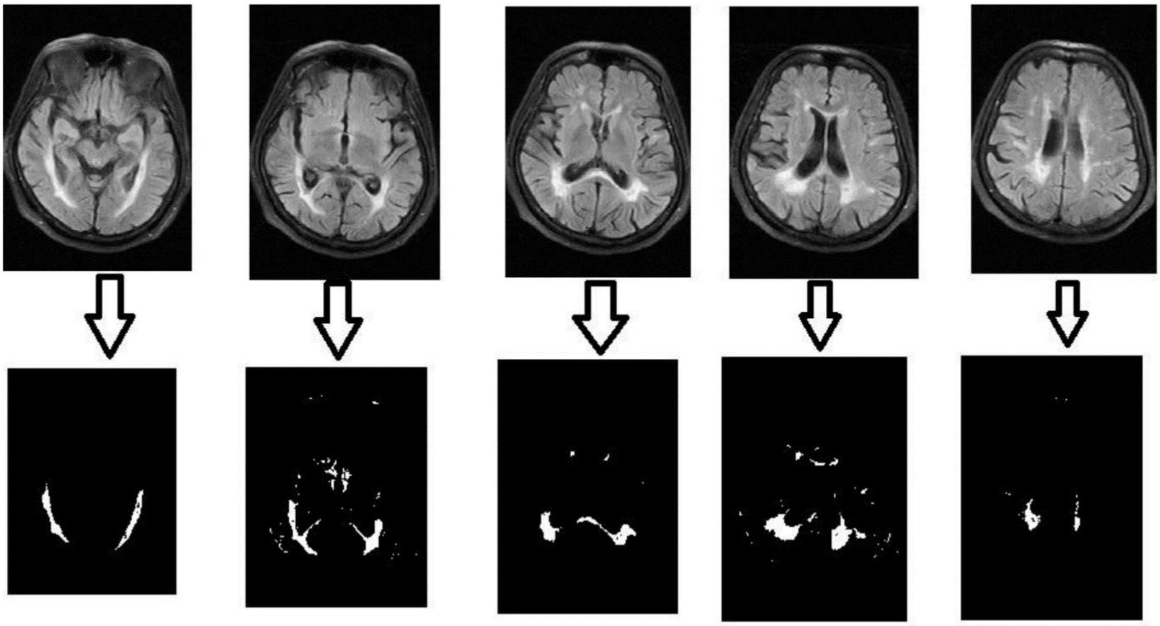


Figure 5-1: The top row is FLAIR images and the bottom line is the initialized Thresholding of the image

### 5-2-2- Finding Central Points (Lesion Gravity Center)

The center of gravity of each of the MS lesions obtained by applying the thresholding algorithm is calculated, and each of these centers is presented as an object (primary lesion).

### 5-2-3- Edge Detection

The edge image is rendered in MATLAB using the edge function; it performs the threshold operation by partially deriving data from adjacent pixels and points where the derivative values are maximal, as the edge of the image is known [111, 113].

### 5-2-4- Find the center of MS lesion level in the image

Determining the center of the surface means that this point can be a fixed point relative to the points on the edge. The center of gravity is calculated using the functions available in MATLAB.

### 5-2-5- Apply the Regional Growth algorithm and find out the range of the lesion

Another work used in image segmentation is Regional Growth, which uses uniformity in the area. This procedure begins by selecting one or more of the granular points and is diagnosed by the same number of zones depending on the number of granular points. Then the similarity between the pixels is formulated. In fact, it can be said that this is a measure of the pixels gray value. Two pixels are also called alike if the magnitude difference of their gray values is less than the threshold value. The area growth operation then begins by inspecting and controlling the points that are enclosed within the dotted areas. If a point is sufficiently similar to the grain points, it belongs to the area corresponding to that grain point. For each area, the operation will continue to cover all points in the image. To make more accurate decisions as well as identify the lesion of each object center, we apply the regional growth algorithm to extract the tissue properties of the lesion. In this step, we hand over the potential lesion center we already acquired to the Regional Growth algorithm so that it can provide us with a range of lesions as well as the number of pixels detected as a second stage lesion.

### 5-2-6- Removal of lesions with a short Euclidean distance from the brain (skull) margin

In fact, we remove lesions that are not restricted to the brain or MS disease, such as lesions on the periphery or areas completely removed from the brain. Areas on the margins are both very low in volume or are specific to other diseases such as head trauma and so on (Fig. 5-2).

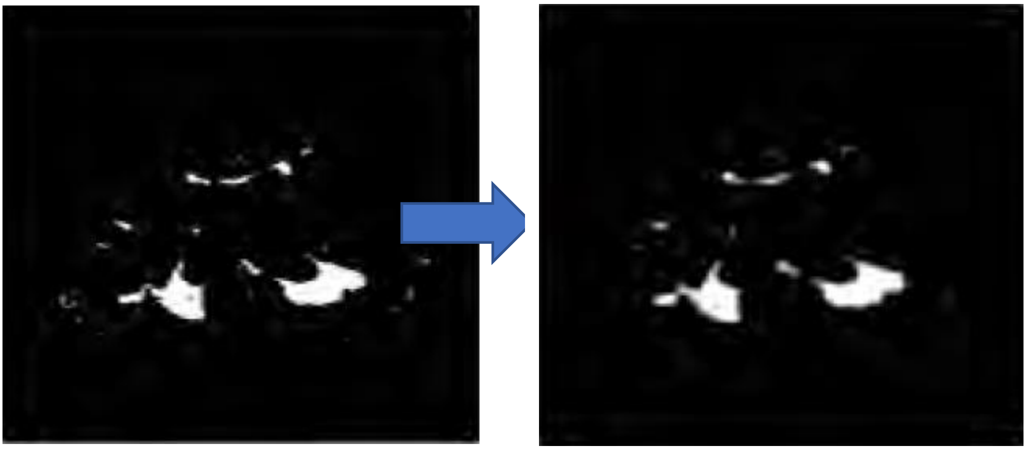


Figure 5-2: Removal of lesions with a short Euclidean distance

### 5-2-7- Removal of lesions less than 4 $mm^3$

Removal of areas less than 4 $mm^3$ in volume is explained by the lack of conformity of low volume lesions with MS and also by the physician to remove low volume areas from the list of lesions. The new initiative we used in this research is to use a graph of points. We divide the zones obtained by applying the regional growth algorithm in each slice into two groups:

- The first group is lesion that has an area of more than 32 pixels and we consider them as lesion pools.
- The second group is a lesion with an area of less than 32 pixels.

The main operations are in relation to the second group. We compare each existing pixel with our 26 neighbors and enter each tree in the growth zone designated as MS area and next to the existing point. Next, for the newly added nodes in the tree, we examine the 26 non-duplicate neighbors and implement exactly the same procedure as before and continue until a new node is added to the tree and if stopped increasing tree masses will be removed from the loop and checked if all slides are not stopped. Next, we count the nodes and make the following decisions according to the following conditions.

1. If the nodules hit a lesion pool but the total nodes are less than 32, we consider them a lesion.
2. If the nodes find a volume greater than 32 nodes, all of the docked together will be considered as MS lesions and the collection will also be recorded as a lesion pool and used for other generations.
3. If one of the above two conditions is not met, these nodes are considered as backgrounds.

Because each voxel in our datasets is 0.5x0.5x0.5 $mm^3$, it requires 8 voxels to form one-millimeter cube, and we need 32 nodes for 4 $mm^3$ (Fig. 5-3).

The difference that we had with other research papers is that we did not consider voxels as a 4 $mm^3$ in three consecutive slides, and we were looking for fitting and bending in the 3D image collection [113].

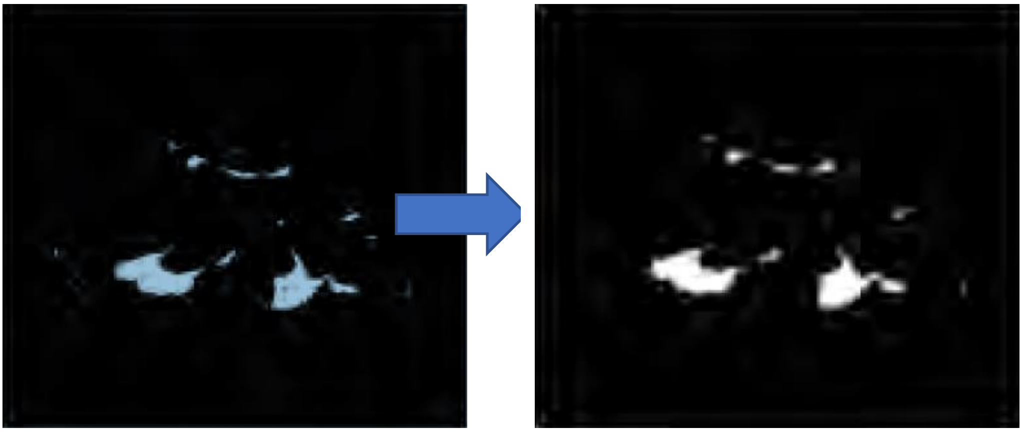


Figure 5-3: Removal of lesions less than 4 mm$^3$

### 5-2-8- Filling holes in the lesion

As can be seen from the medical diagnosis of MS, unlike other diseases, MS is not one-sided and has an internal cavity, so doctors, if present, consider the lesion as the sum of the lesion volume and in the articles [29, 113, 114], We also consider internal cavities (black holes) to be the main lesion and are structurally proportional to the lesion so we fill these holes.

### 5-2-9- Lesion clustering with K-Means algorithm

From the results of the previous step we cluster the lesions in this step using K-Means algorithm, in fact this clustering method separates the lesion points into sets so that in each set of data points to the center of the cluster they're close. The K-Means algorithm has a *K* parameter that represents the number of clusters we consider. The K-Means algorithm is as follows and the results are shown in Fig. (5-4).

1. Initialization of k: considers the point of lesions as the primary centers.
2. Repeat
3. k forms the cluster by assigning all points to the nearest lesion center.
4. The centers of each cluster are recalculated.
5. Until the lesion centers change.

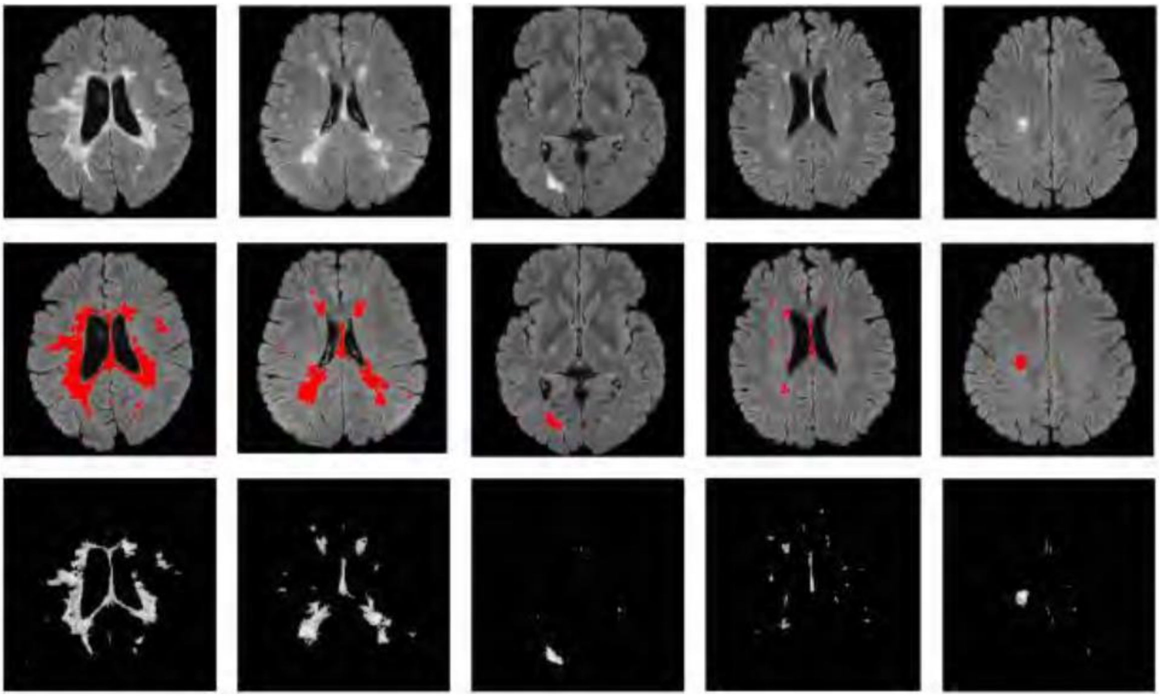


Figure 5-4: The first row shows the original images taken from the brain and the second row shows the segmented lesions and the third row shows the lesions with no other details

The average Dice Similarity Coefficient (DSC) for all MS edges in recent research [115-117] is about 0.891. In this section, DSC = 0.84 is obtained.

## 5-3- Results for Automatic MS Segmentation Approach Based on Cellular Learning Automata

Irregular cellular learning automata (ICLA) are cellular learning automata (CLA) which remove the restriction of rectangular grid structure in traditional CLA. There are number of applications which cannot be adequately modeled with rectangular grids. Each pixel represents a cell that is equipped with a learning automaton. Learning automata are in fact a specific cell that determines its state according to action probability vector [86, 89, 118].

In many matters, learning cannot be correctly operated by using one learning automaton, but the learning automata manifest their main power when employed collectively. Given this issue and the defects enumerated for cellular automata, by combining these two models, a new model called cellular learning automata was generated. To use the algorithm, we first need to have an initial image from the location of the MS plaque (Figure 5-5). The main objective of employing cellular learning automata is making use of neighborhood relations among the pixels of pseudo-skin regions to increase the efficiency [85, 86, 119]. This is because a pixel doesn’t necessarily denote the concept of being a plaque by itself, and a set of pixels can define a skin region in a better way (Fig. 5-5 and 5-6). The penalty and reward of each cell are according to the action chosen by the central automata and its neighbors. If the number of the cells that choose the appropriate action for plaque detection is more than 7, the central automata will be rewarded. Therefore, if seven automata detect a texture as plaque, the region around the central automata is taken as plaque region. If the number of these automata is less than 4 (Fig. 5-6, b, c), the central automata will be penalized [88, 104]. Using databases of the MS images, the images are chosen that have desirable condition regarding the plaques information. For the rate of detection, a formula is used to calculate the percentage error: pixels mistakenly detected as plaques to the total percentage of the detected pixels (Fig. 5-7, c). The aforementioned-phrase is the assessment factor for the percentage error (Figure 5-8), (Table 5-2), and this formula was examined in 40 images of the database and average DSC=0.915 [111].

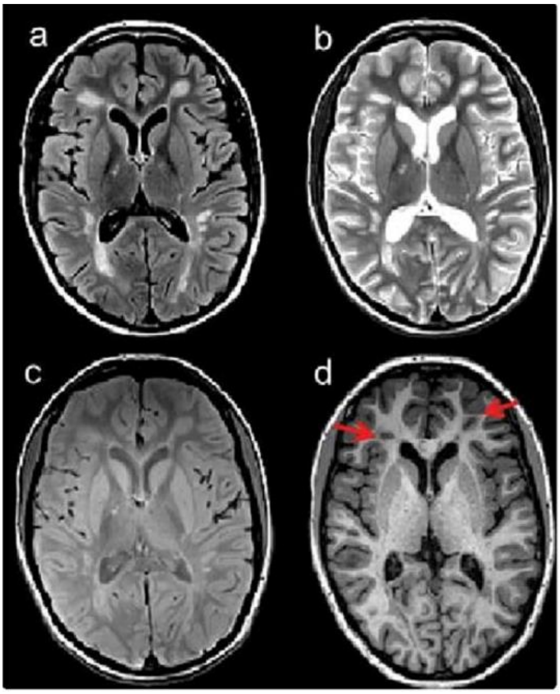
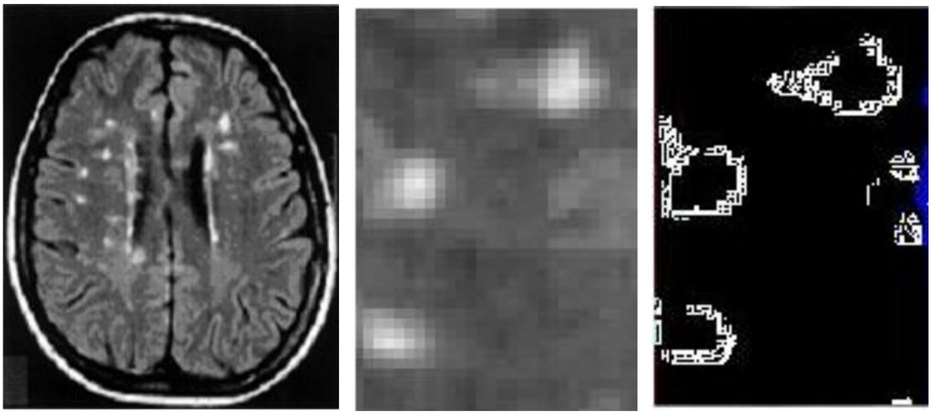
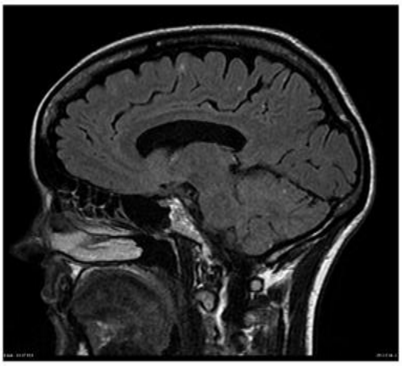
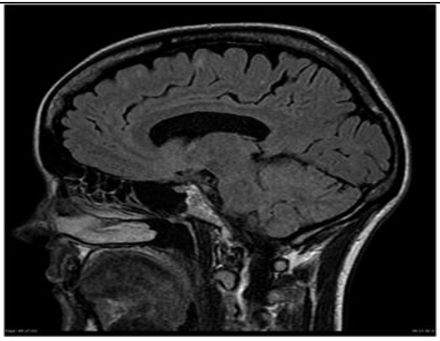
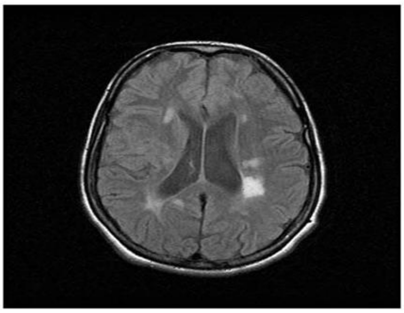


Figure 5-5: MRI sequences depicting MS lesion. Lesions appear bright (hyperintense) on the (a) Fluid-Attenuated Inversion Recovery (FLAIR) Image, (b) T2-Weighted and (c) Proton Density (PD) images. (d) T1 hypointense lesions (black holes) appear dark in the T1-weighted image, with examples indicated by red arrows [144].



(a) (b) (c)

Figure 5-6: a) Image Sample of MS, b) Infected sample, c) The resulted image in MATLAB

| | |
|---|---|
|  |  |
| Image Sample of MS1 plaque | Output image of MS1 plaque using cellular learning automata |
|  |  |
| Image Sample of MS2 plaque | Output of image MS2 plaque using Cellular learning automata |
|  |  |

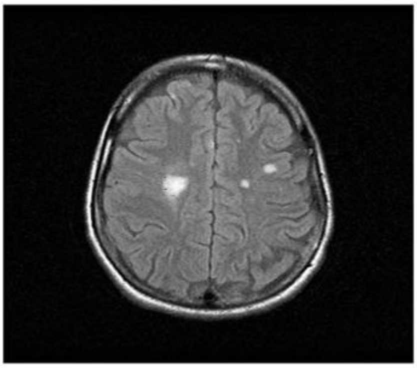
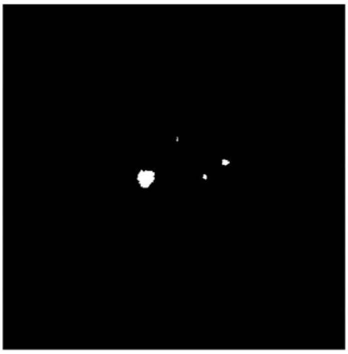
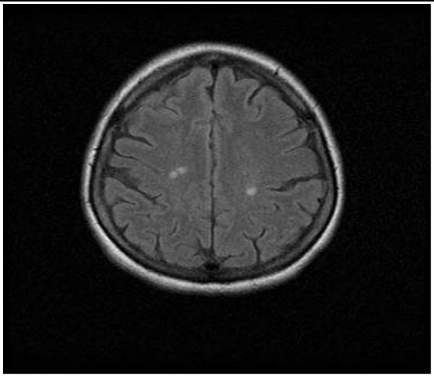
| Image Sample of MS3 plaque | Output of image MS3 plaque using cellular learning automata |
|---|---|
|  |  |
| Image Sample of MS4 plaque | Output of image MS4 plaque using cellular learning automata |
|  |  |
| Image Sample of MS5 plaque | Output of image MS5 plaque using cellular learning automata |

Figure 5-7: MS Images - Analyzed images [111]

Table 5-2: Analyzed images

| Number of the analyzed images | Error rate in finding the pixels that were mistakenly detected as plaques | Error rate in finding the pixels that mistakenly weren't detected as plaques |
|---|---|---|
| 40 images | 10% | 7% |

The pixels depicting MS plaque can be found in the image. By visual inspection, the plaque regions are specified by 1 and other regions are specified by 0. Using ROC command in MATLAB, ROC curve is plotted which is as it is presented below. This curve is used to compare two classification methods. Its Vertical axis is the true positive rate (TPR) and its horizontal axis is the false positive rate (FPR). This curve represents the tradeoff between TPR and FPR for the two-classed classification of MS and healthy plaques. To plot the curve, we assume a threshold; values above the threshold will be considered as "Yes" and values below "No". Conclusively, it is evident that there are various techniques which can be applied in MS plaque detection using the various imaging aspects. From the author’s published research papers, it is clearly obvious that there are various mechanical and operational advantages and disadvantages which emanate from the use of the systems. In addition, the health practitioners need to have a working knowledge of the different systems so that they can be in different positions to adequately deliver as per the required levels. Each technique has its specific advantage and disadvantage and thus, health practitioners need to affirm the best mechanism to set in place based on the gravity of the situation. Primarily, all these aspects can be used in a health institution and it is recommendable for one to have enough background knowledge of the patient before deciding to embark on a specific venture.

## 5-4- Results of MS Segmentation Approach based on SVM and CLA Integration

### 5-4-1- Histogram normalization

Due to the difference in brightness, images are normalized before any operation. Normalization of the intensity of light is essential for the analysis of some tissues in MRI images. According to the results obtained from [69], the histogram normalization method works best, so using this method and by changing the brightness of the original image, the resulting histogram of the brightness of the image contains all gray values. Which is defined as follows:

$$f(x,y) = \frac{GWM - BWM}{h_{max} - h_{min}}(h(x,y) - h_{min}) + BWM \tag{5-1}$$

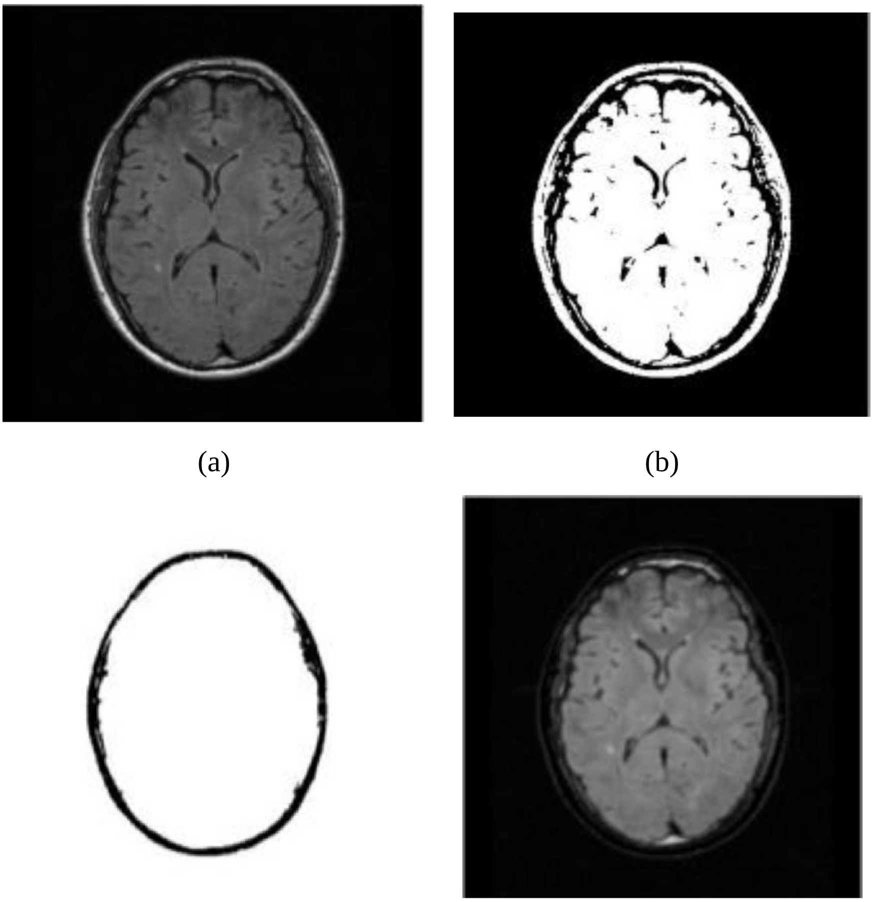
In the above relation $h(x,y)$ histogram of the original image, $f(x,y)$ the new image histogram and $h_{min}$ and $h_{max}$ are the lowest and highest gray values available. GWM and BWM are the lowest and highest brightness intensities of the new image.

### 5-4-2- Skull segmentation

Since noise and image heterogeneity have major effects on non-brain tissue such as skull and fat, it is best to remove non-brain tissue in order to reduce computational time and increase the efficiency of continuous steps. For this purpose, as shown in Fig. (5-8), since in the FLAIR images the skull is highly illuminated, the initial image is initially binary with an appropriate threshold limit. The objects in the binary mask are then numbered in order of their area. This will separate the area of the brain and skull.

### 5-4-3- Detection of lesions with hemispheres

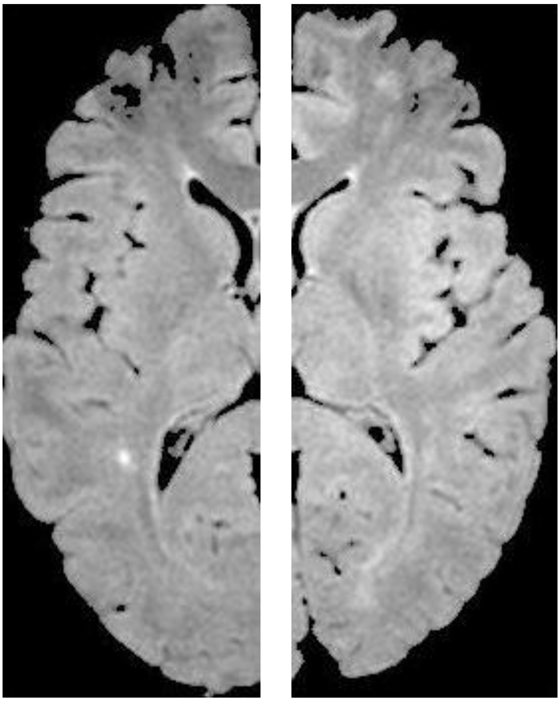
The main idea in the diagnosis of MS lesions is asymmetry of the histogram between healthy and lesions. So, the brain hemispheres must first be extracted. Splitting the brain into two hemispheres is done by finding the diameter of the oval containing the brain. The advantage of this method is that it is resistant to patient rotation and movement during imaging. For this purpose, after complete extraction of the brain, the resulting image becomes binary. The resulting mask has an area that can be separated into two right and left hemispheres by defining the ellipsoid area and considering its large diameter as the midline of the brain [120].



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

Figure 5-8: Skull segmentation. (a) Initial image, (b) binary mask, (c) skull object, (d) final image after skull removal

Also, to reduce the search space and remove the background of the image, remove the black and white areas of the background of the image using the large center and small diameter ellipses (Fig. 5-9).



(a) (b)

Figure 5-9: (a) left hemisphere (b) right hemisphere

### 5-4-4- Feature Classification with SVM

In order to extract the feature, each hemisphere is first swept by a window slider. The distance between the central pixel location of the window and the next location of the pixel is assumed to be 2 pixels to avoid slightly increasing the computation volume. As a result, the window will overlap at the current and next location. In each window the attributes are extracted, and the classifier will be trained to determine whether each training window has a lesion. In the test phase, a window similar to the training phase window is considered, moving across the brain and background area. Features are extracted and classified in this window and wherever it is located. Based on the results of the window center classifications, the mark is based on the location of the lesion or healthy brain tissue. To determine the proper window size, three different windows with sizes of 16x16, 24x24 and 36x36 are considered.

To calculate the histogram, each pixel is mapped to one of the channels based on its gradient angle and casts to that channel. These votes are weighted by the size of the gradient in that pixel. After calculating the gradient histogram in each cell, this histogram is normalized by dividing the sum of the block histograms to which it belongs. Thus, the vector length of the feature extraction property for each block is equal to 32 and for each window 24x24 is equal to 128.

In the training phase, the classification is trained using extractive features and whether or not there is a lesion in each window. In the system evaluation and testing phase, after extracting the features in each window, the central pixel of each window is assigned a zero or one mark according to the presence or absence of lesion in that window as determined by the classification. In order to classify windows a MLTSVM supervised classification method is used. Since the number of healthy windows is much higher than the number of lesion windows which makes the training data not balanced, the number of healthy and patient windows are considered equal and the training data for the healthy windows are considered and selected at random. In the system testing phase to segment the lesion, after extracting the features in each window and classifying it, the central pixel is marked based on the classification result. If the window belongs to the lesion site, its central pixel is set to 0, and if it is to healthy tissue, the value is set to zero. Fig (5-10) shows the results of this method and the extraction lesion contained in the database.

### 5-4-5- Post processing

MS lesions are located in the white area of the brain. In order to increase the accuracy of the system, white matter is extracted using Cellular Learning Automata (CLA) and k-means clustering.

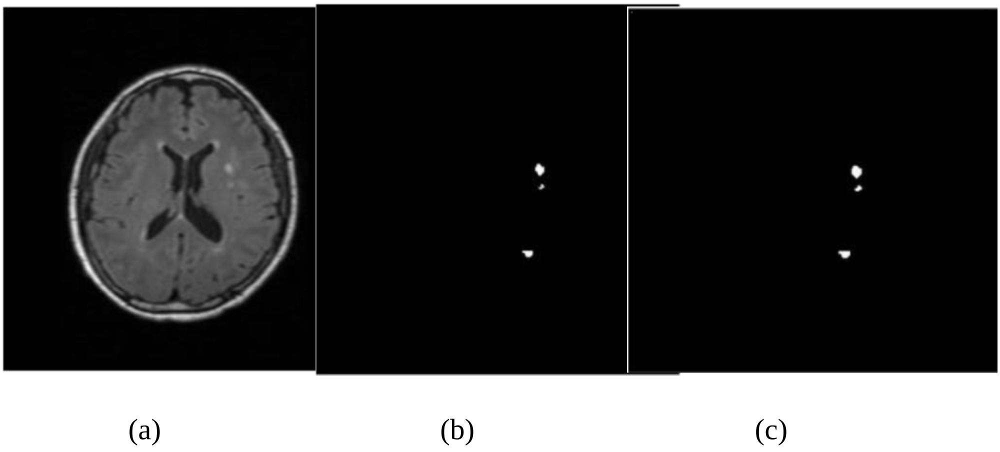

(a) (b) (c)

Figure 5-10- Results of lesion extraction (a) Original image, (b) Histogram extraction lesion and SVM, (c) Database extraction lesion

The lesion is accepted and goes to the next stage, otherwise the lesion is removed Fig (5-11).

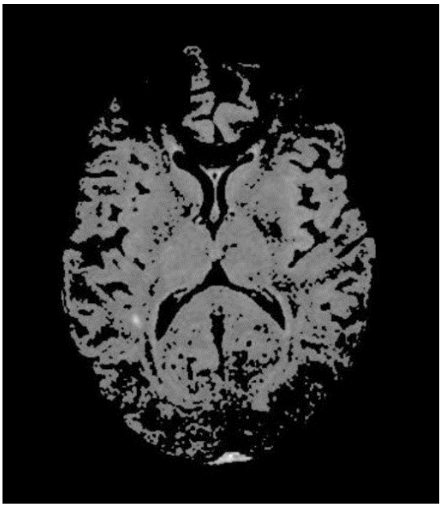


Figure 5-11: Results of clustering by CLA algorithm

### 5-4-6- SVM Results

In the results of the proposed system implementation in order to consider the noise effect, the images used were examined once with noise removal by Gaussian filter and once with no noise removal. The results show that it is better not to eliminate noise because of the loss of some of the contextual information by noise removal (Table 5-3). Therefore, the proposed system is implemented without noise removal. To evaluate the effect of window size, three windows with sizes of 16x16, 24x24 and 36x36 were used and the results are investigated in three classification methods. According to (Table 5-4), the best result is the window size of 24x24 used in the proposed method.

Table 5-3- The effect of noise

| | No noise removal | Noise removal |
|---|---|---|
| SVM Classifier | 95.6 | 87.7 |

Table 5-4- The effect of window size

| | 16x16 | 24x24 | 36x36 |
|---|---|---|---|
| SVM Classifier | 90.6 | 97.3 | 92.4 |

System performance in post-processing modes using CLA with k-means clustering and without its use is shown in (Table 5-5). As can be seen, with the detection of lesioned hemispheres and post-processing with CLA with k-means clustering results in significant improvement. Fig. (5-12) shows the results of the implementation of this system.

Table 5-5- System performance using CLA with k-means clustering and without its use

| | Without CLA with k-means clustering | Using CLA with k-means clustering |
|---|---|---|
| SVM Classifier | 93.2 | 96.4 |

In this method, in this first step, to reduce the computational space, first, based on the histogram of healthy and lesioned hemispheres, the lesioned hemisphere is diagnosed, and then only lesioned hemispheres are examined. An image slider with dimensions of 16x16, 24x24, and 36x36 is used to extract the image's feature and classification. The results show that the most appropriate window size is 24x24.

To investigate the effect of noise, the proposed algorithm is implemented once with noise removal and again without noise removal. The result is better system performance without noise removal. In the post processing step to reduce false positive and false negative error rates since the MS lesion is located in the white area of the brain, this region is extracted by the CLA with k-means clustering and the search space allocated to this region. Based on the results of the study of hemispheres with lesions and CLA with k-means clustering increase system accuracy and performance by 96.4 and average DSC for 36x36 is 0.948.

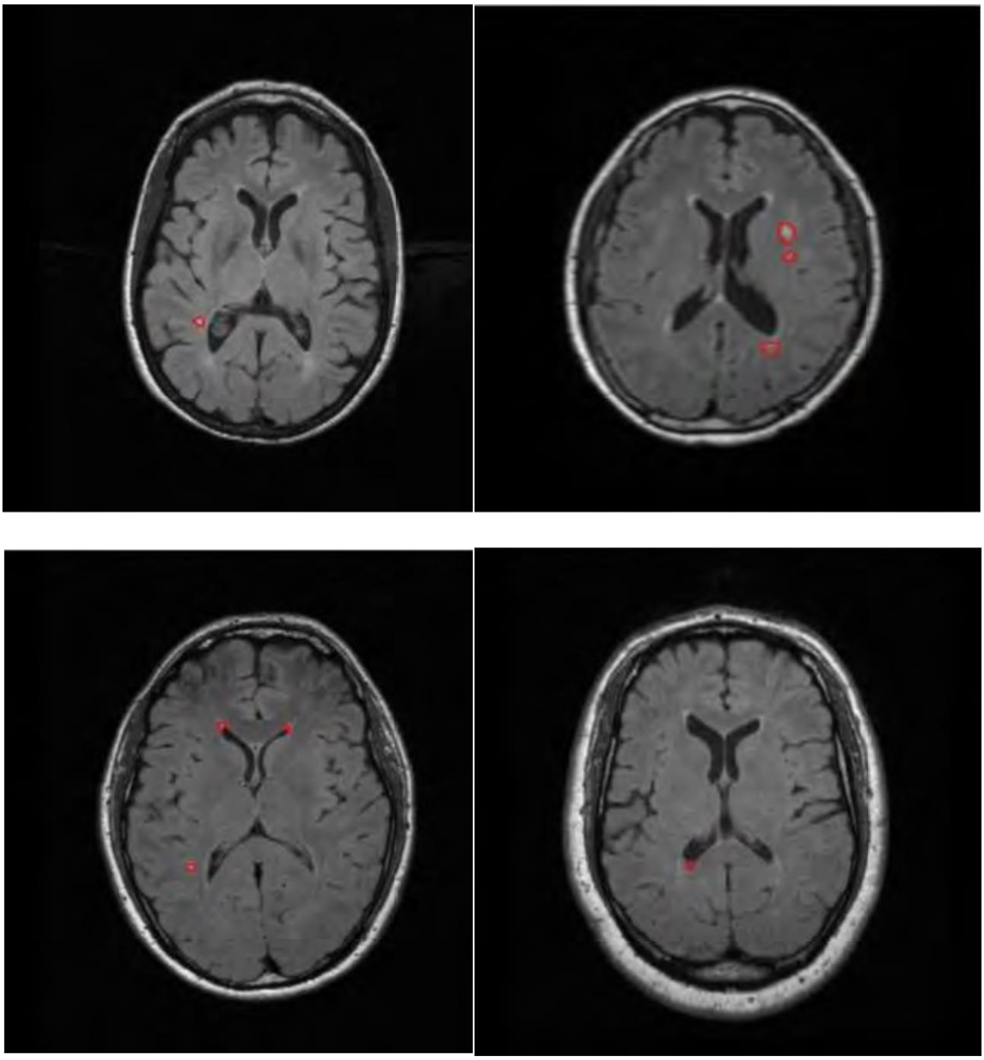


Figure 5-12: Results of lesion extraction by the proposed method on FLAIR images

## 5-5- Results of Simulated Database, 3D MRI to 2D Images

The first step of this section was to determine the appropriate dataset for the machine learning algorithms. For this more source can be appropriated, for instance the dataset used in the below mentioned MS MRI

Lesion Segmentation research. This dataset is the MRI results of 38 MS patients. The disadvantage of this dataset is that it does not contain normal brain MRI images and additional data about the MS severity of the patients [67-70].

Another possible source of the dataset is Brainweb. The Brainweb repository provides a Simulated Brain Database (SBD) containing a set of realistic MRI data volumes generated by an MRI simulator, based on two types of models:

- Normal,
- Multiple Sclerosis (MS),

Brainweb MRI simulator provides full 3-dimensional data volumes, built using three sequences (T1-, T2-, and proton-density- (PD-) weighted) and a variety of slice thicknesses, noise levels, and levels of intensity non-uniformity, available in three orthogonal views (transversal, sagittal, and coronal), and can be downloaded [113]. The main advantages of using a simulated database that extending the dataset is simple and basically without limit, however from simulated images no important meta-information can be processed, for instance age, gender and medical record of the patients in Fig. (5-13), [66, 113, 114].

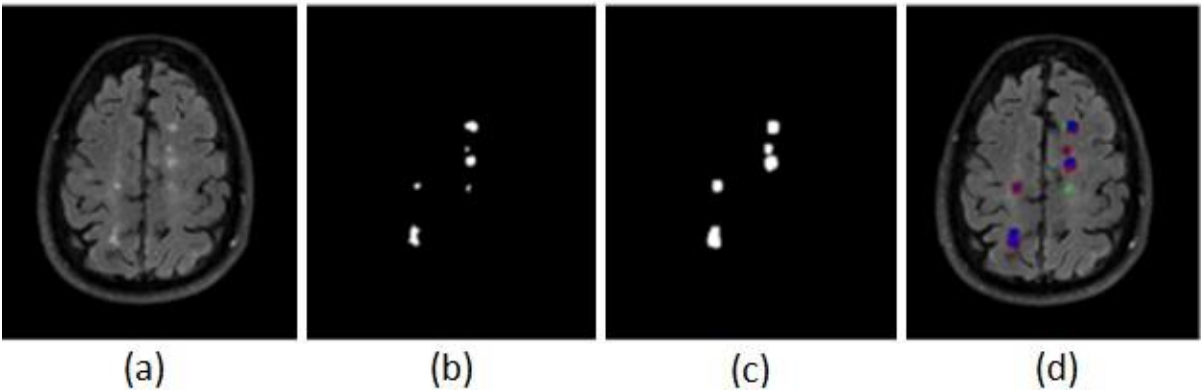


Figure 5-13: BrainWeb: Simulated MRI Volumes for Brain with Multiple Sclerosis Lesions [66, 113, 114]

To experiment with the machine learning algorithms, a simulated dataset can be an acceptable option, since building a real dataset can be time-consuming. Brainweb generates 3D images in binary format [114]. Normal, mild MS, moderate MS and severe MS simulations can be generated using specific parameters mentioned above and three file formats are supported: MINC, unsigned raw byte and 12 bits raw short. In this study 2D images are eligible for the plague detection, since the goal is to create cellular automatas which are able to differentiate between normal and MS brain MRI images. From the 3D dataset, 2D image dataset can be generated, since every layer of the 3D dataset is a 2D image which can be converted into grayscale images. To be able to do this, a MATLAB® script was implemented. This can read and process the 3D dataset and produce the corresponding 2D grayscale images. The dataset acquired from

Brainweb is in MINC format, to be able to read and process this, an additional third-party MATLAB® tool called loadminc written by Laszlo Balkay, must be used [44]. The dimensioning of the elements of original Brainweb dataset is 217x181x181, the dimension of corresponding 2D image is 217x181, and acquired from the z axis of the original dataset using the following logic:

$$d = round\left(\frac{length\ of\ z}{2}\right) \tag{5-2}$$

With the script a dataset containing 72 Images has been generated, each with the following parameters:

- MR pulse sequence:
    - Slice thickness: 1
    - Scan technique: SFLASH
    - Repetition time: 18
    - Inversion time: not used
    - Flip angle: 30
    - Echo time: 10
    - Image Type: magnitude
- Imaging artifacts:
    - Noise reference tissue: brigtest_tissue
    - Noise level (%): 3
    - Random generator seed: generated
    - INU field: field A
    - INU ("RF") level (%): 20

The dataset can be segmented into four classes:

- normal brain,
- brain with mild MS,
- brain with moderate MS,
- brain with severe MS,

Each containing 18 samples (Figures 5-13 & 5-14).

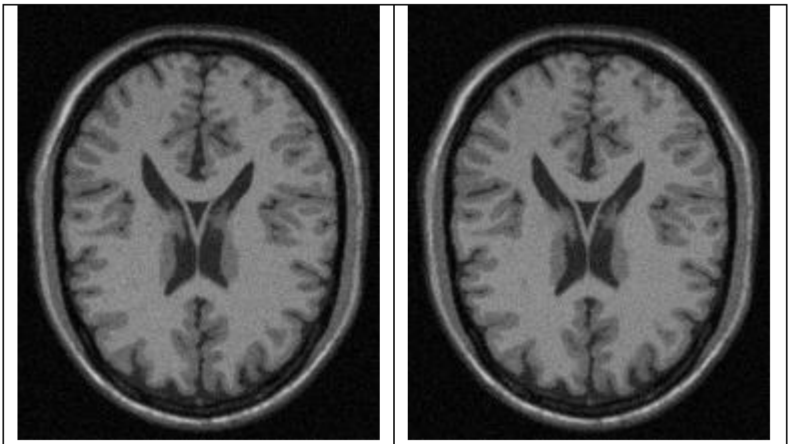


Figure 5-13- Generated images - Brainweb Simulation, Generated normal, mild MS images

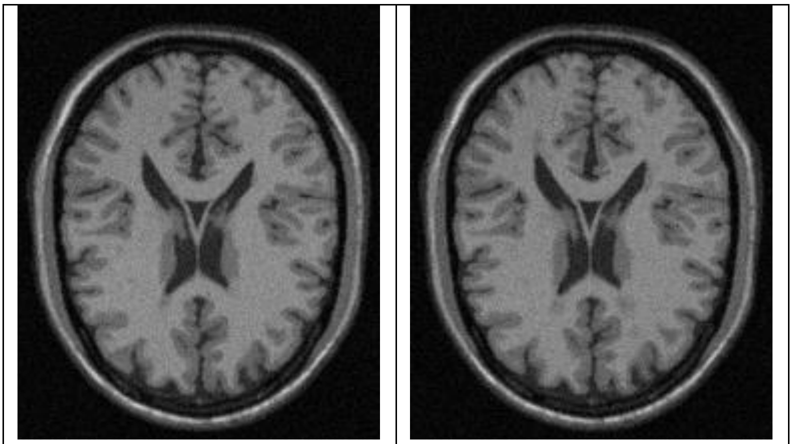


Figure 5-14- Generated images - Brainweb Simulation, Generated moderate MS, Severe MS

### 5-5-1- Results of using SVM

The most common use of SVM is binary pattern classification. SVM aims to construct a hyperplane set in an infinite dimensional space and find the hyperplane which represents the largest separation (margin) between the binary classes, so the goal is to find the maximum-margin hyperplane if it exists [114]. SVM is a binary classifier, however in the current case four mutually exclusive classes can be found in the dataset. To resolve this, One-Against-One (1A1) and One-Against-All (1AA) are commonly used techniques. The 1AA divides the N class dataset into N two-class cases, while 1A1 approach creates a model for each pair of classes so N(N-1)/2 models are built. In these methods every classification method gives one vote to one class and the final class is determined by the votes. Weights can be involved too in the voting process. The disadvantage of the 1AA approach is its performance which can be compromised because of the unbalanced training datasets [128-130]. With 1A1 approach the main disadvantage is the computation intensivity (the condition of being intensive) due to the number of SVM pairs [131].

For building the model MATLAB® fitcsvm function was used with linear kernel function with standardized predictor data. With each method 10 runs were done, in each run 30% of the dataset (23 images) was selected randomly and used for model testing. The analysis of the results is different, it depends on the model used for training [122].

With using 1AA technique, four models were built called Normal2All, MildMS2All, ModerateMS2All and SevereMS2All. Assume that all models have the same weigh during voting the result of the classification can be seen in column 2AllResults. This run is interesting because there are samples which have multiply results (row 4, 8) or no result at all (row 10). In this case no class is assigned to the sample and this rate is also counted in the comparison of the methods (Table 5-6).

Using 1A1 technique six models were built called Normal2MildMS, Normal2ModerateMS, Normal2SevereMS, MildMS2ModerateMS, MildMS2SevereMS and ModerateMS2SevereMS. Assume that all models have the same weigh during voting the result of the classification can be seen in column 2OneResults. During this run, a class has been assigned to every sample, however this is not the case in every run, though the number of cases with multiple result or no result are less compared to the 1AA technique (Table 5-7) [113, 114].

Table 5-6- Example result of One-Against-All classification [113]

| Expected | Normal2All | MildMS2All | ModerateMS2All | SevereMS2All | 2AllResult |
|---|---|---|---|---|---|
| normal | normal | all | all | all | normal |
| normal | normal | all | all | all | normal |
| normal | normal | all | all | all | normal |
| normal | all | mild_ms | all | severe_ms | more_results |
| normal | normal | all | all | all | normal |
| normal | normal | all | all | all | normal |
| normal | normal | all | all | all | normal |
| normal | normal | all | all | severe_ms | more_results |
| mild_ms | normal | all | all | all | normal |
| mild_ms | all | all | all | all | all |
| mild_ms | normal | all | all | all | normal |
| mild_ms | normal | all | all | all | normal |
| mild_ms | all | mild_ms | all | all | mild_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| moderate_ms | all | all | all | severe_ms | severe_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| moderate_ms | all | all | moderate_ms | all | moderate_ms |
| severe_ms | all | all | all | severe_ms | severe_ms |
| severe_ms | all | all | all | severe_ms | severe_ms |
| severe_ms | normal | all | all | all | normal |

Table 5-7- Example result of One-Against-One classification [113]

| Expected | Normal2 MildMS | Normal2 ModerateMS | Normal2 SevereMS | MildMS2 ModerateMS | MildMS2 SevereMS | ModerateMS2 SevereMS | 2OneResult |
|---|---|---|---|---|---|---|---|
| normal | normal | normal | normal | moderate_ms | severe_ms | severe_ms | normal |
| normal | normal | normal | normal | mild_ms | mild_ms | severe_ms | normal |
| normal | normal | normal | normal | mild_ms | severe_ms | severe_ms | normal |
| normal | mild_ms | normal | severe_ms | mild_ms | severe_ms | severe_ms | severe_ms |
| normal | normal | normal | normal | mild_ms | mild_ms | severe_ms | normal |
| normal | normal | normal | normal | mild_ms | mild_ms | severe_ms | normal |
| normal | normal | normal | normal | mild_ms | mild_ms | severe_ms | normal |
| normal | normal | normal | severe_ms | mild_ms | severe_ms | severe_ms | severe_ms |
| mild_ms | normal | normal | normal | moderate_ms | mild_ms | severe_ms | normal |
| mild_ms | normal | normal | severe_ms | mild_ms | severe_ms | severe_ms | severe_ms |
| mild_ms | normal | normal | normal | moderate_ms | mild_ms | severe_ms | normal |
| mild_ms | normal | normal | normal | moderate_ms | mild_ms | severe_ms | normal |
| mild_ms | mild_ms | normal | normal | mild_ms | mild_ms | severe_ms | mild_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| moderate_ms | normal | normal | severe_ms | mild_ms | severe_ms | severe_ms | severe_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| moderate_ms | normal | moderate_ms | severe_ms | moderate_ms | severe_ms | moderate_ms | moderate_ms |
| severe_ms | normal | normal | severe_ms | mild_ms | severe_ms | severe_ms | severe_ms |
| severe_ms | normal | normal | severe_ms | moderate_ms | severe_ms | severe_ms | severe_ms |
| severe_ms | normal | normal | normal | moderate_ms | severe_ms | severe_ms | normal |

To summarize the efficiency of the two techniques of SVM according to the result of the 10 runs (Table 5-8), the 1AA (2AllResult) reached a slightly better average accuracy than 1A1 (2OneResult) and average Dice similarity coefficient (DSC) for 1A1 (2OneResult) is 0.739.

Table 5-8- Overall results for SVM using 1AA and 1A1 technique [113]

| ID | 2AllResult Accuracy | 2OneResult Accuracy | 2AllMore Result | 2OneMore Result | 2AllNo Result | 2OneNo Result | Differences |
|---|---|---|---|---|---|---|---|
| 1 | 0.739130435 | 0.739130435 | 1 | 0 | 0 | 0 | 0.130434783 |
| 2 | 0.826086957 | 0.826086957 | 0 | 0 | 1 | 0 | 0.043478261 |
| 3 | 0.739130435 | 0.739130435 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0.913043478 | 0.913043478 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0.826086957 | 0.782608696 | 2 | 0 | 2 | 0 | 0.391304348 |
| 6 | 0.826086957 | 0.826086957 | 1 | 0 | 2 | 0 | 0.130434783 |
| 7 | 0.782608696 | 0.826086957 | 0 | 0 | 0 | 0 | 0.043478261 |
| 8 | 0.52173913 | 0.434782609 | 2 | 7 | 9 | 0 | 0.565217391 |
| 9 | 0.956521739 | 0.913043478 | 0 | 0 | 0 | 0 | 0.086956522 |
| 10 | 0.652173913 | 0.652173913 | 2 | 0 | 1 | 0 | 0.130434783 |
| **AVG** | 0.77826087 | 0.765217391 | 0.8 | 0.7 | 1.5 | 0 | **0.152173913** |

In all cases the accuracy of the 1A1 approach was the same or worse than the 1AA method. However, comparing the two techniques according to successfully produced result, 1AA technique led to more undefined result with average number of 0.8 when more results were produced and 1.5 when no result was assigned to the sample. 1A1 technique produced more results at an average 0.7 times. Using 1A1 technique producing no result is impossible since the decision is always between two existing classes from the original sample, while in the case of 1AA there is an umbrella class for all samples from the N-1 classes. Producing more result is the same for both methods, if the same amount of vote arrives for the same class, that means that more results were produced. In this case a certain logic can be set in the future, for instance the more severe class can be accepted for the sample or in case of a real dataset other information, like gender, age, symptoms can be added to the model. In case of 1AA no result is produced when all votes from the models come from the 'All' category. The last column 'Differences' from (Table 5-8) summarizes the rate of different answers produced by 1AA and 1A1 techniques for the same sample. This means, that when 1AA produced a 'normal' answer and 1A1 produced 'mild_ms', it is considered a difference therefore it is counted here. This rate shows that the accuracy of the models and the different answers produced by the models do not correlate [113]. As mentioned, SVM is one of the most widely used machine learning method for binary pattern classification. SVM aims to construct a hyperplane set in an infinite dimensional space and find the hyperplane which represents the largest separation (margin) between the binary classes, so the goal is to find the maximum-margin hyperplane if it exists (Chao and Horng, 2015) [127]. In the previous study two approaches were used since SVM is a binary classifier, however the current dataset can be divided into four mutually exclusive classes. In order to resolve this, in the previous study One-Against-One (1A1) and One-Against-All (1AA) techniques were introduced. The goal of 1AA technique to divide the N class dataset into N two-class cases, while 1A1 approach creates a model for each pair of classes so N(N-1)/2 models are built. In the previous study each method had an equal vote. These approaches were reused for the new, fix dataset and to be able to compare the results of the methods the implementation was rerun to build models using the new trainsets and test-sets. For building the model MATLAB® fitcsvm function was used with linear kernel function and with standardized predictor data (Table 5-9)

The average accuracy of the 1AA model is slightly worse compared to the previous model, however the 1A1 model produced a slightly better result. For the 2AllResult (the accuracy of the 1AA models), the previous average accuracy of the models was 0.77826087 while for 2OneResult (the accuracy of the 1A1 models) models was 0.765217391. Therefore, this run, 2OneResult has a slightly better average result and so far, it produced the best results in this run. Another interesting fact that in these runs no sample has been assigned to the ‘more_results’ flag, so in this case with equal votes for each method resulted in a deterministic result, however the number of no results in the case of 1AA method has a significant grow. In the previous published paper, a rule for a voting system could be determined, which could have been tested in this study [114].

Table 5-9- SVM results using linear kernel function

| *ID* | *2All Result* | *2One Result* | *2AllMore Result* | *2OneMore Result* | *2AllNo Result* | *2OneNo Result* | *Differences* | *ST. DEV* |
|---|---|---|---|---|---|---|---|---|
| *1* | *0.739130435* | *0.652173913* | *0* | *0* | *6* | *0* | *0.391304* | *0.061488* |
| *2* | *0.739130435* | *0.782608696* | *0* | *0* | *6* | *0* | *0.26087* | *0.030744* |
| *3* | *0.695652174* | *0.739130435* | *0* | *0* | *7* | *0* | *0.304348* | *0.030744* |
| *4* | *0.826086957* | *0.826086957* | *0* | *0* | *4* | *0* | *0.173913* | *0* |
| *5* | *0.782608696* | *0.782608696* | *0* | *0* | *5* | *0* | *0.26087* | *0* |
| *6* | *0.739130435* | *0.782608696* | *0* | *0* | *6* | *0* | *0.26087* | *0.030744* |
| *7* | *0.739130435* | *0.739130435* | *0* | *0* | *6* | *0* | *0.26087* | *0* |
| *8* | *0.695652174* | *0.75* | *0* | *0* | *7* | *0* | *0.304348* | *0.03843* |
| *9* | *0.782608696* | *0.826086957* | *0* | *0* | *5* | *0* | *0.217391* | *0.030744* |
| *10* | *0.695652174* | *0.782608696* | *0* | *0* | *7* | *0* | *0.304348* | *0.061488* |
| ***AVG*** | ***0.743478261*** | ***0.766304348*** | ***0*** | ***0*** | ***5.9*** | ***0*** | ***0.273913*** | ***0.01614*** |

This was not explained in the previous study, but the system would have been the following:

- 1AA method is used for the primary decision.
- if 1AA resulted in more results, the more severe result should be used.
- if 1AA resulted in no results, the 1A1 result should be used.
- if 1A1 has more results, the more severe should be used.

Table 5-10- SVM results using rbf kernel function

| ID | 2All Result | 2One Result | 2AllMore Result | 2OneMore Result | 2AllNo Result | 2OneNo Result | Differences | ST. DEV |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.739130435 | 0.782608696 | 0 | 0 | 6 | 0 | 0.260869565 | 0.030744 |
| 2 | 0.739130435 | 0.782608696 | 0 | 0 | 6 | 0 | 0.260869565 | 0.030744 |
| 3 | 0.695652174 | 0.739130435 | 0 | 0 | 7 | 0 | 0.304347826 | 0.030744 |
| 4 | 0.826086957 | 0.826086957 | 0 | 0 | 4 | 0 | 0.173913043 | 0 |
| 5 | 0.782608696 | 0.826086957 | 0 | 0 | 5 | 0 | 0.217391304 | 0.030744 |
| 6 | 0.739130435 | 0.739130435 | 0 | 0 | 6 | 0 | 0.260869565 | 0 |
| 7 | 0.739130435 | 0.739130435 | 0 | 0 | 6 | 0 | 0.260869565 | 0 |
| 8 | 0.695652174 | 0.782608696 | 0 | 0 | 7 | 0 | 0.304347826 | 0.061488 |
| 9 | 0.782608696 | 0.826086957 | 0 | 0 | 5 | 0 | 0.217391304 | 0.030744 |
| 10 | 0.695652174 | 0.739130435 | 0 | 0 | 7 | 0 | 0.304347826 | 0.030744 |
| **AVG** | **0.743478261** | **0.778260870** | **0** | **0** | **5.9** | **0** | **0.256521739** | 0.024595 |

This rule should be rejected, because of the 1A1 has produced a slightly better result compared to the 1AA method and because no sample got more results in this case, so using such a technique wouldn't improve the overall results. For SVM, a different kernel function can be tested to examine the method efficiency [114]. This kernel function is the radial basis function (rbf) and the result can be seen in (Table 5-10).

A neural network (NN) is built from simple, connected processors called neurons which form a sequence of real-valued activations. Special type of neurons, called input neurons are activated through the environment, other neurons are activated through the connected previously activated, weighted neurons. The learning process of a neural network is finding the optimal weights to be able to perform the desired results (Table 5-10). The construction of a network depends on the problem and how the neurons are connected [132]. MATLAB® provides easy to use, flexible tools to create, test and analyze neural networks. In this study a simple artificial neural network (for the exact parameters see Appendix A), with 15 layers and 10 max epochs were trained using the sample train- and test datasets which were used to build the SVM models [113, 114].

# Chapter 6

## Conclusion, Suggestions and Future Works

## 6-1- Conclusion

This dissertation practices the research of SVM used for classification and CLA, then it expands the research to other method such as k-NN and ANN and at that point compares the results of them. 3D images can be converted into 2D and by considering machine learning techniques and SVM tools. The work concludes that SVM is a potential method which can be involved during MS diagnosis, however, in order to confirm this statement more research and other potentially effective methods should be included and need to be tested. And a combined algorithm for better MS lesion diagnosis was proposed. In the feature extraction and feature selection stage, texture, shape, and severity features were extracted from MS lesions. A number of these features play a more important role in lesion diagnosis and were selected for more accurate and effective classification. Figure (6-1) shows the competency function. Counting matrix was used to classify and evaluate its performance in detecting healthy tissue and Ms tissue from MRI images of the brain. For this purpose, the features extracted from the texture, shape and intensity of the image were used to prepare the samples detected by the support vector machine. The classification performance was evaluated for accuracy, precision, sensitivity and MSE. Accuracy, precision, classification sensitivity with support vector machine in this study is 100% or zero MSE. The results of the feature selection classification function and the support vector machine are shown in Table 6-1.

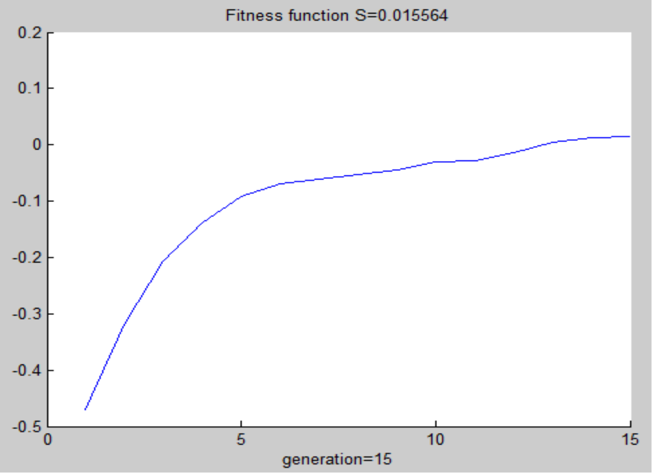


Figure 6-1- competency function for feature extraction and feature selection

Table 6-1- Performance classification performance results Select feature and backup vector machine

| Incorrect forecast | Correct prediction | MSE | Specificity | Sensitivity | Accuracy |
|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 1 |

Table 6-2- results of segmentation performance

|  | Overlap | Similarity | FN<br>False Negative | FP<br>False Positive |
|---|---|---|---|---|
| Pictures | 0.8595 | 1 | 0.0081 | 0.0106 |

The accuracy of segmentation (in terms of overlap and similarity) of the output image with the real image was measured and evaluated based on the application of logical operators on binary images. Table (6-2) shows the results of segmentation performance. In this study two approaches were used since SVM is a binary classifier, however the current dataset can be divided into four mutually exclusive classes.

A) Integrated semi-automatic method and K-Means for detect MS

An attractive and accurate alternative to manual segmentation is computer segmentation. This can be done more accurately and in less time for the practitioners. To make it easier for the doctors to make a decision, the information obtained can be clustered and a better picture of the disease can be provided. Using the K-means algorithm, we clustered the division results, which made it much easier for the specialist doctors to separate the lesions. The Dice Similarity Coefficient (DSC) = 0.84 result shows that the algorithm is reliable and can be used in clinics and medical centers for MS.

B) Using Cellular Learning Automata (CLA) for automatic MS segmentation

The segmentation of the lesions from one another is studied by using gray scale features and the dimensions of the lesions. The brain Magnetic Resonance Imaging (MRI) images in three planes T1, T2, PD (weighted) containing MS disease lesions have been used. Cellular Learning Automata (CLA) is applied on the MRI images with a novel trial and error approach to set penalty and reward frames for each pixel and average DSC = 0.915.

C) Integrated SVM, K-Means and CLA for segmentation of MS lesions

In this first step, to reduce the computational space, first, based on the histogram of healthy and lesioned hemispheres, the lesioned hemisphere is diagnosed, and then only lesioned hemispheres are examined. An image slider with dimensions of 16x16, 24x24, and 36x36 is used to extract the image's feature and classification. The results show that the most appropriate window size is 24x24. To investigate the effect of noise, the proposed algorithm is implemented once with noise removal and again without noise removal. The result is better system performance without noise removal. In the post processing step to reduce false positive and false negative error rates since the MS lesion is located in the white area of the brain, this region is extracted by the CLA with k-means clustering and the search space allocated to this region. Based on the results of the study of hemispheres with lesions and CLA with k-means clustering increase system accuracy and performance by 96.4 and average DSC for 36x36 is 0.948.

D) Using value of binary pattern classification to MS detection from simulated 3D MRI to 2D Images.

To be able to make better assumptions, more tests are needed. Using Brainweb, images can be generated with different parameters, which can change the parameters (contrast, brightness etc.) of the images. In this study, only 2D images were generated from 3D images, one by one, however for each 3D image more 2D images can be generated using different axis or layer. Even models using the whole 3D dataset can be built, if the proper hardware resources are available. The 1AA (2AllResult) reached a slightly better average accuracy than 1A1 (2OneResult) and average DSC for 1A1 (2OneResult) is 0.939.

Comparison of the results of different MS lesion detection methods is shown in Figure (6-1), and the results show that integrated SVM, K-Means and CLA method (DSC=0.948) has better results than DSC = 0.891 from previous research [115-117].

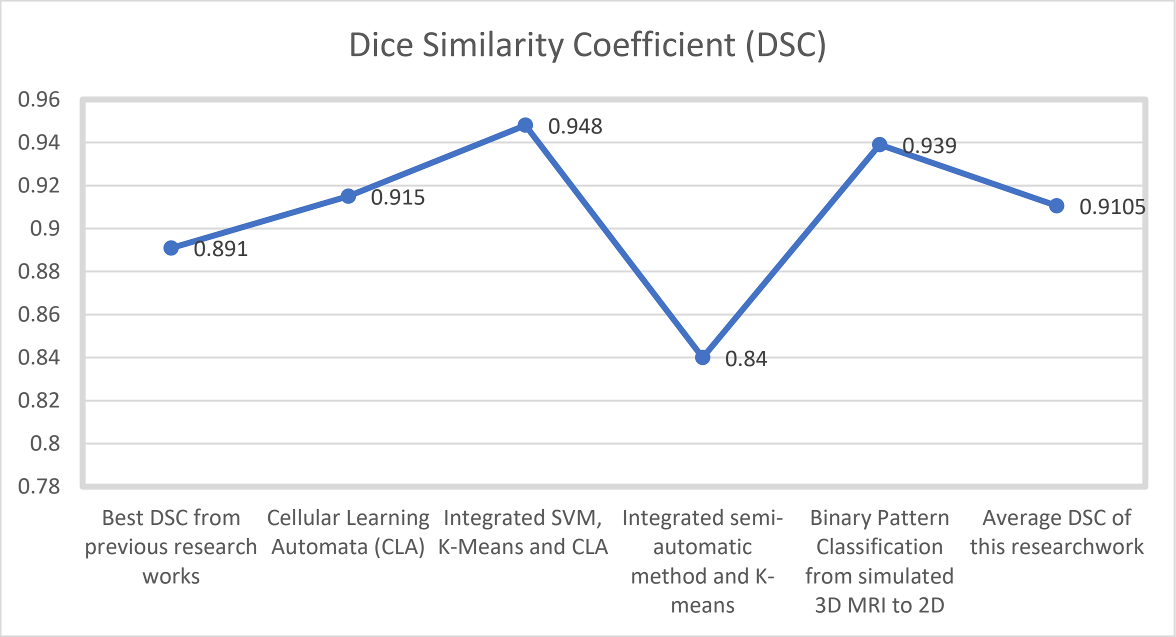


Figure 6-1- Comparison of accuracy of different methods for diagnosing MS

## 6-2 Suggestions and future works

Our main focus in this study was on suggesting a novel machine learning and classification algorithm to improve MS detection in MRI images and to improve the performance of computational methods. Some issues related to MS lesion detection and available algorithm, their advantage and disadvantages need further research and have not been addressed in this thesis.

### 6-2-1 Design a mechanism for deep, layered and fast learning

Deep learning is introduced as a new concept and a subset of artificial intelligence and uses neural networks to improve machine behaviors such as speech recognition, computer vision, and natural language processing. Deep learning simulates high-level abstract concepts by learning at different levels and layers of a neural network. Neural networks have an inner hidden layer, and any neural network that has multiple dermal layers is called the deep network [132]. Deep learning is usually used for images of cannabis networks and for textual data such as natural language deep recursive networks, supervised learning, unsupervised learning and semi-supervised learning [133-135]. In semi-supervised learning, unlabeled data and labeled data are used simultaneously to improve learning accuracy. In order to control the depth of learning as well as its speed, an approach is suggested that the supervised learning departments operate independently without supervision. In other words, the learning process for the labeled and unlabeled data is carried out in parallel and the learning algorithms of the latest researches of the researchers are adjusted for each section and the Fuzzy Min-Max Neural Network (FMNN) is used to further the learning because of the aforementioned method to enhance learning. It is that in the

MRI reports and the hospital information system, the share of unsupervised learning is greater than that of supervised learning, yet both types of data are involved in the final decision making.

### 6-2-2 Design of medical decision support system

Designing a medical decision support system for early detection of MS is not only dependent on MRI reports and the hospital information system, but physicians may need other patient information to make the final decision, so we need to have a mechanism in place. Both patient information can be easily searched for in a user-friendly environment, using its decision-making system, and other patient information can be reviewed if needed. Designers of a medical decision support system have been designed by various researchers, for example Latifi and Ayed in a study in 2016 suggested the use of an intuitive medical decision support system in the medical field and examined the challenges of designing such a system [136], the overview of the research of these researchers is shown in Figure (6-2).

In a general perspective, the work of researchers in this field is to visualize information. Information visualization is considered as a way of managing large sets of scientific data and improving the ability to discover relationships and phenomena in data. Visualization is a process in which data, information and knowledge are transformed visually and utilize individuals' natural abilities to quickly recognize the visual pattern. Effective user interfaces enable us to view, manipulate, search, browse, discover, filter, identify, understand, and interact effectively with data to discover hidden patterns. The impact of visualization is widespread and fundamental, leading to new attitudes and effective decisions [137].

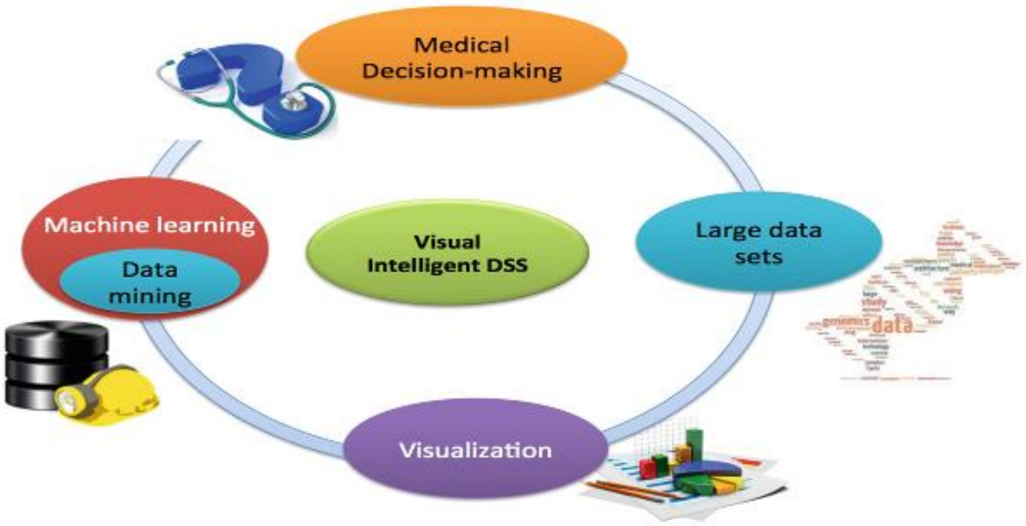


Figure 6-2: Suggested workflow for intuitive intelligent decision support system

Visualization is a new approach that enables the ability to make data-based decisions faster, more accurately, and with less cognitive effort and reduces the need for training to understand information [138]. The visualization design approach can be divided into several steps. Chittaro draws a six-step list, including: mapping, selection, presentation, interactivity, usability, and evaluation that illustrate major visualization activities for less accurate and error-prone design [139]. One of the areas related to visualization is data visualization, which is graphical representation designed to support the viewer in a qualitative understanding of the information content. Information may be data, processes, relationships, or concepts. [137-140].

# References

[1] M. T Jock Murray, Multiple sclerosis: the history of a disease. Demos medical publishing, 2004.

[2] F. Bennett, D. C. Clemmons, and R. T. Fraser, Multiple sclerosis: Psychosocial and vocational interventions. Demos Medical Publishing, 2002.

[3] R. T. Fraser, G. H. Kraft, D. M. Ehde, and K. L. Johnson, "The MS workbook: Living fully with multiple sclerosis," 2006.

[4] M. Filippi, C. Tortorella, and M. Bozzali, "Normal-appearing white matter changes in multiple sclerosis: the contribution of magnetic resonance techniques," Multiple Sclerosis Journal, vol. 5, no. 4, pp. 273-282, 1999.

[5] B. Johnston, M. S. Atkins, B. Mackiewich, and M. Anderson, "Segmentation of multiple sclerosis lesions in intensity corrected multispectral MRI," IEEE Transactions on Medical Imaging, vol. 15, no. 2, pp. 154-169, 1996.

[6] R. E. Sakai, D. J. Feller, K. M. Galetta, S. L. Galetta, and L. J. Balcer, "Vision in multiple sclerosis (MS): The story, structure-function correlations, and models for neuroprotection," Journal of neuro-ophthalmology: the official journal of the North American Neuro-Ophthalmology Society, vol. 31, no. 4, p. 362, 2011.

[7] D. García-Lorenzo, S. Francis, S. Narayanan, D. L. Arnold, and D. L. Collins, "Review of automatic segmentation methods of multiple sclerosis white matter lesions on conventional magnetic resonance imaging," Medical image analysis, vol. 17, no. 1, pp. 1-18, 2013.

[8] K. Pandey et al., "Clinical course in multiple sclerosis patients presenting with a history of progressive disease," Multiple sclerosis and related disorders, vol. 3, no. 1, pp. 67-71, 2014.

[9] A. Ortiz, J. Górriz, J. Ramírez, D. Salas-Gonzalez, and J. M. Llamas-Elvira, "Two fully-unsupervised methods for MR brain image segmentation using SOM-based strategies," Applied Soft Computing, vol. 13, no. 5, pp. 2668-2682, 2013.

[10] A. Cerasa et al., "A cellular neural network methodology for the automated segmentation of multiple sclerosis lesions," Journal of neuroscience methods, vol. 203, no. 1, pp. 193-199, 2012.

[11] K. Bendfeldt et al., "Multivariate pattern classification of gray matter pathology in multiple sclerosis," Neuroimage, vol. 60, no. 1, pp. 400-408, 2012.

[12] S. Klöppel et al., "Accuracy of dementia diagnosis—a direct comparison between radiologists and a computerized method," Brain, vol. 131, no. 11, pp. 2969-2974, 2008.

[13] C. Elliott, D. L. Arnold, D. L. Collins, and T. Arbel, "Temporally consistent probabilistic detection of new multiple sclerosis lesions in brain MRI," IEEE transactions on medical imaging, vol. 32, no. 8, pp. 1490-1503, 2013.

[14] N. Shiee, P.-L. Bazin, A. Ozturk, D. S. Reich, P. A. Calabresi, and D. L. Pham, "A topology-preserving approach to the segmentation of brain images with multiple sclerosis lesions," NeuroImage, vol. 49, no. 2, pp. 1524-1535, 2010.

[15] A. Ortiz, J. M. Gorriz, J. Ramírez, D. Salas-Gonzalez, and A. s. D. N. Initiative, "Improving MRI segmentation with probabilistic GHSOM and multiobjective optimization," Neurocomputing, vol. 114, pp. 118-131, 2013.

[16] P. Anbeek, K. L. Vincken, M. J. Van Osch, R. H. Bisschops, and J. Van Der Grond, "Probabilistic segmentation of white matter lesions in MR imaging," NeuroImage, vol. 21, no. 3, pp. 1037-1044, 2004.

[17] T. Logeswari and M. Karnan, "An improved implementation of brain tumor detection using segmentation based on hierarchical self organizing map," International Journal of Computer Theory and Engineering, vol. 2, no. 4, p. 591, 2010.

[18] S. Klöppel et al., "A comparison of different automated methods for the detection of white matter lesions in MRI data," NeuroImage, vol. 57, no. 2, pp. 416-422, 2011.

[19] J. Bogaert, S. Dymarkowski, A. M. Taylor, and V. Muthurangu, Clinical cardiac MRI. Springer Science & Business Media, 2012.

[20] W. Schempp and W. Schempp, Magnetic resonance imaging: Mathematical foundations and applications. Wiley-Liss New York, 1998.

[21] R. Failo, P. A. Wielopolski, H. A. Tiddens, W. C. Hop, R. Pozzi Mucelli, and M. H. Lequin, "Lung morphology assessment using MRI: a robust ultra-short TR/TE 2D steady state free precession sequence used in cystic fibrosis patients," Magnetic Resonance in Medicine: An Official

Journal of the International Society for Magnetic Resonance in Medicine, vol. 61, no. 2, pp. 299-306, 2009.

[22] M. Van Walderveen et al., "Correlating MRI and clinical disease activity in multiple sclerosis: relevance of hypointense lesions on short-TR/short-TE (T1-weighted) spin-echo images," Neurology, vol. 45, no. 9, pp. 1684-1690, 1995.

[23] T. Yokoo et al., "Nonalcoholic fatty liver disease: diagnostic and fat-grading accuracy of low-flip-angle multiecho gradient-recalled-echo MR imaging at 1.5 T," Radiology, vol. 251, no. 1, pp. 67-76, 2009.

[24] M. Cabezas et al., "Automatic multiple sclerosis lesion detection in brain MRI by FLAIR thresholding," Computer methods and programs in biomedicine, vol. 115, no. 3, pp. 147-161, 2014.

[25] X. Lladó et al., "Segmentation of multiple sclerosis lesions in brain MRI: a review of automated approaches," Information Sciences, vol. 186, no. 1, pp. 164-185, 2012.

[26] A. Jog, A. Carass, D. L. Pham, and J. L. Prince, "Random forest FLAIR reconstruction from T1, T2, and PD-weighted MRI," in 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), 2014, pp. 1079-1082: IEEE.

[27] J. K. Udupa, L. Wei, S. Samarasekera, Y. Miki, M. A. van Buchem, and R. I. Grossman, "Multiple sclerosis lesion quantification using fuzzy-connectedness principles," IEEE Transactions on medical imaging, vol. 16, no. 5, pp. 598-609, 1997.

[28] B. A. Abdullah, A. A. Younis, P. M. Pattany, and E. Saraf-Lavi, "Textural based SVM for MS lesion segmentation in FLAIR MRIs," Open Journal of Medical Imaging, vol. 1, no. 2, p. 26, 2011.

[29] D. Mortazavi, A. Z. Kouzani, and H. Soltanian-Zadeh, "Segmentation of multiple sclerosis lesions in MR images: a review," Neuroradiology, vol. 54, no. 4, pp. 299-320, 2012.

[30] L. Palm Dzung, X. Chenyang, and L. Prince Jerry, "A survey of current methods in medical image segmentation," 1998: Technical report. Johns Hopkins University, Baltimore, 1998.

[31] D. García-Lorenzo, S. Prima, L. Parkes, J.-C. Ferré, S. P. Morrissey, and C. Barillot, "The impact of processing workflow in performance of automatic white matter lesion segmentation in multiple sclerosis," in MICCAI workshop on Medical Image Analysis on Multiple Sclerosis (validation and methodological issues) (MIAMS'2008), 2008, pp. 104-112.

[32] R. H. Hashemi, W. G. Bradley, and C. J. Lisanti, MRI: The Basics: The Basics. Lippincott Williams & Wilkins, 2012.

[33] D. S. David and J. P. Sutton, "Towards automated enhancement, segmentation and classification of digital brain images using networks of networks," Information Sciences, vol. 138, no. 1-4, pp. 45-77, 2001.

[34] M. B. Cuadra, L. Cammoun, T. Butz, O. Cuisenaire, and J.-P. Thiran, "Comparison and validation of tissue modelization and statistical classification methods in T1-weighted MR brain images," IEEE transactions on medical imaging, vol. 24, no. 12, pp. 1548-1565, 2005.

[35] P. Gibbs, D. L. Buckley, S. J. Blackband, and A. Horsman, "Tumour volume determination from MR images by morphological segmentation," Physics in Medicine & Biology, vol. 41, no. 11, p. 2437, 1996.

[36] W. E. Polakowski et al., "Computer-aided breast cancer detection and diagnosis of masses using difference of Gaussians and derivative-based feature saliency," IEEE transactions on medical imaging, vol. 16, no. 6, pp. 811-819, 1997.

[37] N. Senthilkumaran and R. Rajesh, "Image segmentation-a survey of soft computing approaches," in 2009 International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 844-846: IEEE.

[38] M. Kaya, "Image clustering and compression using an annealed fuzzy hopfield neural network," international journal of signal processing, vol. 80, 2005.

[39] M. Abdulghafour, "Image segmentation using Fuzzy logic and genetic algorithms," 2003.

[40] X.-B. Wen, H. Zhang, and Z.-T. Jiang, "Multiscale unsupervised segmentation of SAR imagery using the genetic algorithm," Sensors, vol. 8, no. 3, pp. 1704-1711, 2008.

[41] C. O. Bagayoko, H. Müller, and A. Geissbuhler, "Assessment of Internet-based tele-medicine in Africa (the RAFT project)," Computerized Medical Imaging and Graphics, vol. 30, no. 6-7, pp. 407-416, 2006.

[42] N. Shah and S. Mahajan, "Distributed Document Clustering Using K-Means," International Journal, vol. 4, no. 11, 2014.

[43] F. Iglesias and W. Kastner, "Analysis of similarity measures in times series clustering for the discovery of building energy patterns," Energies, vol. 6, no. 2, pp. 579-597, 2013.

[44] A. S. Shirkhorshidi, S. Aghabozorgi, T. Y. Wah, and T. Herawan, "Big data clustering: a review," in International Conference on Computational Science and Its Applications, 2014, pp. 707-720: Springer.

[45] J. A. Silva, E. R. Faria, R. C. Barros, E. R. Hruschka, A. C. De Carvalho, and J. Gama, "Data stream clustering: A survey," ACM Computing Surveys (CSUR), vol. 46, no. 1, p. 13, 2013.

[46] P. P. Anchalia, A. K. Koundinya, and N. Srinath, "MapReduce design of K-means clustering algorithm," in Information Science and Applications (ICISA), 2013 International Conference on, 2013, pp. 1-5: IEEE.

[47] A. D. Barrachina and A. O'Driscoll, "A big data methodology for categorising technical support requests using Hadoop and Mahout," Journal of Big Data, vol. 1, no. 1, p. 1, 2014.

[48] M. Yuwono, S. W. Su, B. Moulton, and H. T. Nguyen, "Data Clustering Using Variants of Rapid Centroid Estimation," IEEE Trans. Evolutionary Computation, vol. 18, no. 3, pp. 366-377, 2014.

[49] A. Chadha and S. Kumar, "An improved K-Means clustering algorithm: A step forward for removal of dependency on K," in Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on, 2014, pp. 136-140: IEEE.

[50] T. Debatty, P. Michiardi, W. Mees, and O. Thonnard, "Determining the k in k-means with MapReduce," 2014.

[51] K. E. Dierckens, A. B. Harrison, C. K. Leung, and A. V. Pind, "A Data Science and Engineering Solution for Fast K-Means Clustering of Big Data," in Trustcom/BigDataSE/ICESS, 2017 IEEE, 2017, pp. 925-932: IEEE.

[52] R. Rouhi and M. Jafari, "Classification of benign and malignant breast tumors based on hybrid level set segmentation," expert systems with Applications, vol. 46, pp. 45-59, 2016.

[53] A. Tirtajaya and D. D. Santika, "Classification of microcalcification using dual-tree complex wavelet transform and support vector machine," in Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference on, 2010, pp. 164-166: IEEE.

[54] J. Dheeba and S. T. Selvi, "Classification of malignant and benign microcalcification using SVM classifier," in Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on, 2011, pp. 686-690: IEEE.

[55] L. Kaufman, "Solving the quadratic programming problem arising in support vector classification," ed: MIT Press, Cambridge, MA, 1998, pp. 147-167.

[56] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998.

[57] V. Vapnik, The nature of statistical learning theory. Springer science & business media, 2013.

[58] Y.-J. Lee and S.-Y. Huang, "Reduced support vector machines: A statistical theory," IEEE Trans. Neural Networks, vol. 18, no. 1, pp. 1-13, 2007.

[59] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," Journal of machine learning research, vol. 1, no. Jun, pp. 211-244, 2001.

[60] D. Tomar and S. Agarwal, "Twin support vector machine: a review from 2007 to 2014," Egyptian Informatics Journal, vol. 16, no. 1, pp. 55-69, 2015.

[61] D. Li and Y. Tian, "Twin support vector machine in linear programs," Journal of Computational Science, vol. 10, pp. 270-277, 2015.

[62] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," Neural processing letters, vol. 9, no. 3, pp. 293-300, 1999.

[63] Y. Xu, L. Wang, and P. Zhong, "A rough margin-based ν-twin support vector machine," Neural Computing and Applications, vol. 21, no. 6, pp. 1307-1317, 2012.

[64] W.-J. Chen, Y.-H. Shao, C.-N. Li, and N.-Y. Deng, "MLTSVM: a novel twin support vector machine to multi-label learning," Pattern Recognition, vol. 52, pp. 61-74, 2016.

[65] L. Farhi, R. Zia, and Z. A. Ali, "5 Performance Analysis of Machine Learning Classifiers for Brain Tumor MR Images," Sir Syed Research Journal of Engineering & Technology, vol. 8, no. 1, pp. 6-6, 2018.

[66] D. L. Collins et al., "Design and construction of a realistic digital brain phantom," IEEE transactions on medical imaging, vol. 17, no. 3, pp. 463-468, 1998.

[67] C. P. Loizou et al., "Brain white matter lesion classification in multiple sclerosis subjects for the prognosis of future disability," Intelligent Decision Technologies, vol. 7, no. 1, pp. 3-10, 2013.

[68] C. P. Loizou, V. Murray, M. S. Pattichis, I. Seimenis, M. Pantziaris, and C. S. Pattichis, "Multiscale amplitude-modulation frequency-modulation (AM–FM) texture analysis of multiple sclerosis in brain MRI images," IEEE Transactions on Information Technology in Biomedicine, vol. 15, no. 1, pp. 119-129, 2010.

[69] C. P. Loizou, M. Pantziaris, I. Seimenis, and C. Pattichis, "Brain MR image normalization in texture analysis of multiple sclerosis," in 2009 9th International Conference on Information Technology and Applications in Biomedicine, 2009, pp. 1-5: IEEE.

[70] C. P. Loizou, S. Petroudi, I. Seimenis, M. Pantziaris, and C. S. Pattichis, "Quantitative texture analysis of brain white matter lesions derived from T2-weighted MR images in MS patients with clinically isolated syndrome," Journal of Neuroradiology, vol. 42, no. 2, pp. 99-114, 2015.

[71] B. A. Abdullah, A. A. Younis, and N. M. John, "Multi-sectional views textural based SVM for MS lesion segmentation in multi-channels MRIs," The open biomedical engineering journal, vol. 6, p. 56, 2012.

[72] R. Kelly, "Critical Comparison of the Classification Ability of Deep Convolutional Neural Network Frameworks with Support Vector Machine Techniques in the Image Classification Process," 2017.

[73] L. I. Komlósi and P. Waldbuesser, "The cognitive entity generation: Emergent properties in social cognition," in 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 439-442: IEEE.

[74] V. Davis and S. Devane, "Diagnosis & classification of brain hemorrhage," in 2017 International Conference on Advances in Computing, Communication and Control (ICAC3), 2017, pp. 1-6: IEEE.

[75] M. Sridevi, S. Naaz, and K. Anjum, "Diagnosis and classification of brain hemorrhage using CAD system," Indian Journal of Scientific Research, pp. 121-126, 2015.

[76] V. Leelavathy and S. S. Kumar, "Optimal design of grading ring and analysis of electric field of 800 kV UHVDC bushing," in 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2015, pp. 1-5: IEEE.

[77] A. de Montaigu and G. Coupland, "The timing of GIGANTEA expression during day/night cycles varies with the geographical origin of

Arabidopsis accessions," Plant signaling & behavior, vol. 12, no. 7, p. e1342026, 2017.

[78] J. Tóth, R. Tornai, I. Labancz, and A. Hajdu, "Efficient Visualization for an Ensemble-based System," Acta Polytechnica Hungarica, vol. 16, no. 2, pp. 59-75, 2019.

[79] J. Von Neumann and A. W. Burks, "Theory of self-reproducing automata," IEEE Transactions on Neural Networks, vol. 5, no. 1, pp. 3-14, 1966.

[80] M. A. Thathachar and P. S. Sastry, "Varieties of learning automata: an overview," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 32, no. 6, pp. 711-722, 2002.

[81] M. Esmaeilpour, V. Naderifar, and Z. Shukur, "Cellular learning automata for mining customer behaviour in shopping activity," International Journal of Innovative Computing, Information, & Control, vol. 8, no. 4, pp. 2491-2511, 2012.

[82] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Foundations and Trends® in Machine learning, vol. 3, no. 1, pp. 1-122, 2011.

[83] H. Beigy and M. R. Meybodi, "Cellular learning automata with multiple learning automata in each cell and its applications," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 40, no. 1, pp. 54-65, 2009.

[84] S. Nascimento, B. Mirkin, and F. Moura-Pires, "A fuzzy clustering model of data and fuzzy c-means," in Ninth IEEE International Conference on Fuzzy Systems. FUZZ-IEEE 2000 (Cat. No. 00CH37063), 2000, vol. 1, pp. 302-307: IEEE.

[85] A. A. Abin, M. Fotouhi, and S. Kasaei, "A new dynamic cellular learning automata-based skin detector," Multimedia systems, vol. 15, no. 5, pp. 309-323, 2009.

[86] H. Beigy and M. R. Meybodi, "Open synchronous cellular learning automata," Advances in Complex Systems, vol. 10, no. 04, pp. 527-556, 2007.

[87] S. M. Vahidipour, M. R. Meybodi, and M. Esnaashari, "Learning automata-based adaptive petri net and its application to priority assignment in queuing systems with unknown parameters," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, no. 10, pp. 1373-1384, 2015.

[88] H. Beigy and M. R. Meybodi, "A mathematical framework for cellular learning automata," Advances in complex systems, vol. 7, no. 03n04, pp. 295-319, 2004.

[89] A. A. Abin, M. Fotouhi, and S. Kasaei, "Skin segmentation based on cellular learning automata," in Proceedings of the 6th international conference on advances in mobile computing and multimedia, 2008, pp. 254-259: ACM.

[90] J. M. Keller, D. Liu, and D. B. Fogel, Fundamentals of computational intelligence: neural networks, fuzzy systems, and evolutionary computation. John Wiley & Sons, 2016.

[91] K. Gurney, An introduction to neural networks. CRC press, 2014.

[92] K. Van Leemput, F. Maes, D. Vandermeulen, A. Colchester, and P. Suetens, "Automated segmentation of multiple sclerosis lesions by model outlier detection," IEEE transactions on medical imaging, vol. 20, no. 8, pp. 677-688, 2001.

[93] R. De Boer et al., "AUTOMATIC SEGMENTATION OF BRAIN TISSUE ANDWHITEMATTER LESIONS IN MRI," in 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2007, pp. 652-655: IEEE.

[94] S. Datta and P. A. Narayana, "A comprehensive approach to the segmentation of multichannel three-dimensional MR brain images in multiple sclerosis," NeuroImage: Clinical, vol. 2, pp. 184-196, 2013.

[95] E. Roura et al., "A toolbox for multiple sclerosis lesion segmentation," Neuroradiology, vol. 57, no. 10, pp. 1031-1043, 2015.

[96] T. Brosch, L. Y. Tang, Y. Yoo, D. K. 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.

[97] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, "Constrained k-means clustering with background knowledge," in Icml, 2001, vol. 1, pp. 577-584.

[98] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," Pattern recognition, vol. 36, no. 2, pp. 451-461, 2003.

[99] R. Shamir and R. Sharan, "1 1 algorithmic approaches to clustering gene expression data," Current Topics in Computational Molecular Biology, p. 269, 2002.

[100] A. Fahim, A. Salem, F. A. Torkey, and M. Ramadan, "An efficient enhanced k-means clustering algorithm," Journal of Zhejiang University-Science A, vol. 7, no. 10, pp. 1626-1633, 2006.

[101] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in Advances in neural information processing systems, 2002, pp. 849-856.

[102] J. Kumar, R. T. Mills, F. M. Hoffman, and W. W. Hargrove, "Parallel k-means clustering for quantitative ecoregion delineation using large data sets," Procedia Computer Science, vol. 4, pp. 1602-1611, 2011.

[103] R. Farivar, D. Rebolledo, E. Chan, and R. H. Campbell, "A Parallel Implementation of K-Means Clustering on GPUs," in Pdpta, 2008, vol. 13, no. 2, pp. 212-312.

[104] H. Beigy and M. R. Meybodi, "Cellular learning automata with multiple learning automata in each cell and its applications," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 40, no. 1, pp. 54-65, 2010.

[105] A. Rezvanian and M. R. Meybodi, "A new learning automata-based sampling algorithm for social networks," International Journal of Communication Systems, vol. 30, no. 5, p. e3091, 2017.

[106] M. D. Heath, S. Sarkar, T. Sanocki, and K. W. Bowyer, "A robust visual method for assessing the relative performance of edge-detection algorithms," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 12, pp. 1338-1359, 1997.

[107] J. Canny, "A computational approach to edge detection," in Readings in computer vision: Elsevier, 1987, pp. 184-203.

[108] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," SIAM Journal on Numerical analysis, vol. 29, no. 1, pp. 182-193, 1992.

[109] B. Moradabadi and M. R. Meybodi, "Wavefront cellular learning automata," Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 28, no. 2, p. 021101, 2018.

[110] S. H. Gheshlaghi, A. Madani, A. Suratgar, and F. Faraji, "Segmentation of Multiple Sclerosis lesion in brain MR images using Fuzzy C-Means," arXiv preprint arXiv:1804.03282, 2018.

[111] M. Moghadasi and D. Gabor, "An Automatic Multiple Sclerosis Lesion Segmentation Approach based on Cellular Learning Automata,"

International Journal of Advanced Computer Science and Applications, vol. 10, 01/01 2019.

[112] J. R. Miller, "The importance of early diagnosis of multiple sclerosis," Journal of managed care pharmacy, vol. 10, no. 3, p. S4, 2004.

[113] M. Moghadasi and G. Fazekas, "Multiple Sclerosis Detection via Machine Learning Algorithm, Accurate Simulated Database 3D MRI to 2D Images, using value of Binary Pattern Classification - A Case Study," 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2019, pp. 233-240, doi: 10.1109/CogInfoCom47531.2019.9089962

[114] Mohammad Moghadasi and Gabor Fazekas, "Multiple sclerosis Lesion Detection via Machine Learning Algorithm based on converting 3D to 2D MRI images", Infocommunications Journal, Vol. XII, No 1, March 2020, pp. 38-44. http://dx.doi.org/10.36244/ICJ.2020.1.6

[115] M. Weeda et al., "Validation of mean upper cervical cord area (MUCCA) measurement techniques in multiple sclerosis (MS): High reproducibility and robustness to lesions, but large software and scanner effects," vol. 24, p. 101962, 2019.

[116] P. Ghosal, P. K. C. Prasad, and D. Nandi, "A Light Weighted Deep Learning Framework for Multiple Sclerosis Lesion Segmentation," in 2019 Fifth International Conference on Image Information Processing (ICIIP), 2019, pp. 526-531: IEEE.

[117] K. Buyukturkoglu, E. Mormina, P. L. De Jager, C. S. Riley, and V. M. J. J. o. N. Leavitt, "The Impact of MRI T1 Hypointense Brain Lesions on Cerebral Deep Gray Matter Volume Measures in Multiple Sclerosis," vol. 29, no. 4, pp. 458-462, 2019.

[118] T. A. Woolsey, J. Hanaway, and M. H. Gado, The brain atlas: a visual guide to the human central nervous system. John Wiley & Sons, 2017.

[119] M. Ahangaran, N. Taghizadeh, and H. Beigy, "Associative cellular learning automata and its applications," Applied Soft Computing, vol. 53, pp. 1-18, 2017.

[120] N. Nabizadeh and M. Kubat, "Brain tumors detection and segmentation in MR images: Gabor wavelet vs. statistical features," Computers & Electrical Engineering, vol. 45, pp. 286-301, 2015.

[121] I. Horváth, "Disruptive technologies in higher education," in 2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2016, pp. 000347-000352: IEEE.

[122] M. U. b. database. Available: http://www.bic.mni.mcgill.ca/brainweb/

[123] R.-S. Kwan, A. C. Evans, and G. B. Pike, "MRI simulation-based evaluation of image-processing and classification methods," IEEE transactions on medical imaging, vol. 18, no. 11, pp. 1085-1097, 1999.

[124] R. K.-S. Kwan, A. C. Evans, and G. B. Pike, "An extensible MRI simulator for post-processing evaluation," in International Conference on Visualization in Biomedical Computing, 1996, pp. 135-140: Springer.

[125] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, G. B. Pike, and A. C. Evans, "Brainweb: Online interface to a 3D MRI simulated brain database," in NeuroImage, 1997: Citeseer.

[126] A. Hajdu, J. Toth, Z. Pistar, B. Domokos, and Z. Torok, "An ensemble-based collaborative framework to support customized user needs," in 2012 IEEE 3rd International Conference on Cognitive Infocommunications (CogInfoCom), 2012, pp. 285-290: IEEE.

[127] C.-F. Chao and M.-H. Horng, "The construction of support vector machine classifier using the firefly algorithm," Computational intelligence and neuroscience, vol. 2015, p. 2, 2015.

[128] S. M. Mousavi, M. Moghadasi, and G. Fazekas, "Dynamic resource allocation using combinatorial methods in Cloud: A case study," in 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 000073-000078: IEEE.

[129] G. Csapó, "Sprego virtual collaboration space," in 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 000137-000142: IEEE.

[130] V. Rentoumi et al., "Automatic detection of linguistic indicators as a means of early detection of Alzheimer's disease and of related dementias: A computational linguistics analysis," in 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 000033-000038: IEEE.

[131] G. Anthony, H. Gregg, and M. Tshilidzi, "Image classification using SVMs: one-against-one vs one-against-all," arXiv preprint arXiv:0711.2914, 2007.

[132] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural networks, vol. 61, pp. 85-117, 2015.

[133] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, pp. 436-444, 2015.

[134] L. Deng and D. Yu, "Deep learning: methods and applications," Foundations and Trends® in Signal Processing, vol. 7, no. 3–4, pp. 197-387, 2014.

[135] Mohammad Moghadasi, Seyed Majid Mousavi and Gábor Fazekas, “Cloud Computing Auditing” International Journal of Advanced Computer Science and Applications(ijacsa), 9(12), 2018. http://dx.doi.org/10.14569/IJACSA.2018.091265

[136] H. Ltifi and M. B. Ayed, "Visual Intelligent Decision Support Systems in the Medical Field: Design and Evaluation," in Machine Learning for Health Informatics: Springer, 2016, pp. 243-258.

[137] N. Gershon and S. G. Eick, "Guest editors' introduction: Information visualization. The next frontier," Journal of Intelligent Information Systems, vol. 11, no. 3, pp. 199-204, 1998.

[138] J. Agutter and J. Bermudez, "Information visualization design: The growing challenges of a data saturated world," AIA Report on university research, pp. 61-75, 2005.

[139] L. Chittaro, "Visualizing information on mobile devices," Computer, vol. 39, no. 3, pp. 40-45, 2006.

[140] M. Chau, "Visualizing web search results using glyphs: Design and evaluation of a flower metaphor," ACM Transactions on Management Information Systems (TMIS), vol. 2, no. 1, p. 2, 2011.

[141] D. Yamamoto et al., "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, vol. 34, no. 5, pp. 404-413, 2010.

[142] R. C. Gonzales and R. E. Woods, "Digital image processing," ed: Prentice hall New Jersey, 2002.

[143] X. J. Zhu, "Semi-supervised learning literature survey," University of Wisconsin-Madison Department of Computer Sciences2005.

[144] Akbar N, Rudko DA, Parmar K. Magnetic Resonance Imaging of Multiple Sclerosis. Sci J Mult Scler. 2017; 1(1): 008-020.

# Publications

## *List of Related publications*

1. **M. Moghadasi** and Gabor Fazekas, "Multiple sclerosis Lesion Detection via Machine Learning Algorithm based on converting 3D to 2D MRI images", Infocommunications Journal, Vol. XII, No 1, March 2020, pp. 38-44. http://dx.doi.org/10.36244/ICJ.2020.1.6
2. **M. Moghadasi** and D. Gabor, "An Automatic Multiple Sclerosis Lesion Segmentation Approach based on Cellular Learning Automata," International Journal of Advanced Computer Science and Applications, vol. 10, 01/01 2019.
3. **M. Moghadasi** and G. Fazekas, "Multiple Sclerosis Detection via Machine Learning Algorithm, Accurate Simulated Database 3D MRI to 2D Images, using value of Binary Pattern Classification - A Case Study," 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2019, pp. 233-240, doi: 10.1109/CogInfoCom47531.2019.9089962
4. **M. Moghadasi**, Seyed Majid Mousavi and Gábor Fazekas, “Cloud Computing Auditing” International Journal of Advanced Computer Science and Applications(ijacsa), 9(12), 2018. http://dx.doi.org/10.14569/IJACSA.2018.091265
5. **M. Moghadasi,** S. M. Mousavi and G. Fazekas, "Dynamic resource allocation using combinatorial methods in Cloud: A case study," in 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 000073-000078: IEEE.
6. **M. Moghadasi,** M. R. Amini, and I. Fatehi, "A BFSK Neural Network Demodulator with Fast Training Hints," *2010 Second International*

*Conference on Communication Software and Networks*, Singapore, 2010, pp. 578-582, doi: 10.1109/ICCSN.2010.123.

*Further publications*

7. **M. Moghadasi** and Dr. G. Fazekas, " Multiple sclerosis Lesion Detection via Machine Learning Algorithm based on converting 3D to 2D MRI images," Under Review Paper, Special Issue of the Infocommunication Journal, "Cognitive Infocommunications Theory and Applications" (Submitted).

8. **M. Moghadasi** and Dr. G. Fazekas, " The Division of Multiple Sclerosis Lesions into Brain MRI Images Using a Combination of Machine Learning Methods and Geometric Features," Under Review Paper, Elsevier Journal, jvci - Journal of Visual Communication and Image Representation (Submitted).

9. **M. Moghadasi** and Dr. G. Fazekas, " An Automatic Multiple Sclerosis Lesion Segmentation Approach based on SVM and Cellular Learning Automata Clustering," Under Review Paper, CogInfoCom Special Issue in Acta Polytechnica Hungarica, "Digital transformation environment for education" (Submitted).

10. **M. Moghadasi** and Dr. G. Fazekas, " Classification and diagnosis of multiple sclerosis lesion in brain MRI images using clustering and genetic algorithm," Under Review Paper, CogInfocom Special Issue in Future Internet, Future Internet "Cognitive Infocommunications–Theory and Applications" (Submitted).

# Appendix A

```
layers = [
    imageInputLayer([217 181 1])

convolution2dLayer(3,8,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2)

convolution2dLayer(3,16,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2)

convolution2dLayer(3,32,'Padding','same')
    batchNormalizationLayer
    reluLayer
    fullyConnectedLayer(4)
    softmaxLayer
    classificationLayer];
options = trainingOptions('sgdm', ...
    'InitialLearnRate',0.01. ...
    'MaxEpochs',50. ...
    'Shuffle','every-epoch', ...
    'Verbose',false);
```