University of Mumbai

Brain Tumor Detection and Classification

Submitted in partial fulfillment of requirements

For the degree of

Bachelors in Technology

by

Apurva Mehta

Hitesh Jaiswal

Shreyansh Kotak

Yash Pasar

Guide

Prof. Poonam Bhogle

Department of Computer Engineering
K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

Batch 2014 -2018

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate

This is to certify that the dissertation report entitled **Brain Tumor Detection and Classification** is bona fide record of the dissertation work done by Apurva Mehta, Hitesh Jaiswal, Shreyansh Kotak and Yash Pasar in the year 2017-18 under the guidance of Prof. Poonam Bhogle of Department of Computer Engineering in partial fulfillment of requirement for the Bachelors in Computer Technology degree in Computer Engineering of University of Mumbai.

__________________ ______________________

Guide Head of the Department

____

Principal

Date:

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate of Approval of Examiners

We certify that this dissertation report entitled **Brain Tumor Detection and Classification** is bona fide record of project work done by Apurva Mehta, Hitesh Jaiswal, Shreyansh Kotak and Yash Pasar.

This project is approved for the award of Bachelors in Technology Degree in Computer Engineering of University of Mumbai.

Internal Examiner

External Examiner

Date:

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

DECLARATION

We declare that this written thesis submission represents the work done based on my and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

____________________ **Signature of the Student** ____________________ **Roll No.**	____________________ **Signature of the Student** ____________________ **Roll No.**
____________________ **Signature of the Student** ____________________ **Roll No.**	____________________ **Signature of the Student** ____________________ **Roll No.**

Date:

Place: Mumbai-77

Abstract

The brain is the most important organ in the human body, responsible for controlling and regulating all critical life functions for the body and a tumor is a mass of tissue formed by the accumulation of abnormal cells, which keep on growing. A brain tumor is a tumor which is either formed in the brain or has migrated No primary cause has been identified for the formation of tumors in the brain till date. Though tumors in the brain are not very common (Worldwide brain tumors make up only 1.8% of total reported tumors), the mortality rate of malignant brain tumors is very high due to the fact that the tumor formation is in the most critical organ of the body. Hence, it is of utmost importance to accurately detect brain tumors at early stages to lower the mortality rate.

We have thus proposed a computer-assisted radiology system which will assess brain tumors from MRI scans for the management of brain tumor diagnosis. In this study, we have implemented a model that segments images using Watershed and PSO algorithm, extracts features using DWT and PCA algorithms and finally classifies the tumors using CNN, SVM and Lazy IBK algorithms with a high degree of accuracy.

Contents

List of Figures

List of Tables

Chapter 1. Introduction

This chapter discusses the main concepts that the project is based on. It identifies what the project is actually meant to accomplish.

1.1 Background

The human body consists of myriad number of cells. When cell growth becomes uncontrollable the extra mass of cell transforms into tumor. CT scans and MRI scans are used for identification of the tumor. The goal of our study is to accurately detect tumors in the brain and classify it through the means of several techniques involving medical image processing, pattern analysis, and computer vision for enhancement, segmentation and classification of brain diagnosis. This system can be used by neurosurgeons, radiologists and healthcare specialists. The system is expected to improve the sensitivity, specificity, and diagnostic efficiency of brain tumor screening using industry standard simulation software tool, MATLAB. These techniques involve pre-processing of MRI scans collected from online cancer imaging archives as well as scans obtained from several pathology labs. Images are resized and then we apply the proposed algorithms for segmentation and classification. The system is expected to improve the brain tumor screening procedure currently at use, and possibly reduce health care costs by decreasing the need for follow-up procedures. Several processing steps are required for the accurate characterization and analysis of biomedical image data.

1.2 Problem Definition

Our study deals with automated brain tumor detection and classification. Normally the anatomy of the brain is analyzed by MRI scans or CT scans. Our system aims to detect if the given MRI scan has a tumor and if found it then classifies the tumor as malignant or benign.

1.3 Motivation

The motivation of the proposed application is to aid neurosurgeons and radiologists in detecting brain tumors in an inexpensive and non-invasive manner.

1.4 Scope

Our aim is to develop an automated system for enhancement, segmentation and classification of brain tumors. The system can be used by neurosurgeons and healthcare specialists. The system incorporates image processing, pattern analysis, and computer vision techniques and is expected to improve the sensitivity, specificity, and efficiency of brain tumor screening. The proper combination and parameterization of above phases enables the development of adjunct tools that can help on the early diagnosis or the monitoring of the therapeutic procedures.

Chapter 2. Literature Survey

This chapter discusses the papers referenced in preparation for undertaking this project. These papers serve as a benchmark to enable this project to be undertaken.

In Medical diagnosis, robustness and accuracy of the prediction algorithms are very important, because the result is crucial for treatment of patients. There are many popular classification and clustering algorithms used for prediction. The goal of clustering a medical image is to simplify the representation of an image into a meaningful image and make it easier to analyze. Several Clustering and Classification algorithms are aimed at enhancing the prediction accuracy of diagnosis process in detecting abnormalities.

The first section of this chapter presents different pre-processing techniques on the MRI Images obtained. Types of Segmentation algorithms used are given in the second section. The next section discusses about the different feature extraction methods and finally the classification algorithms have been presented followed by the summary of this chapter.

2.1 Pre – Processing and Enhancement

Preprocessing and enhancement techniques are used to improve the detection of the suspicious region from Magnetic Resonance Image (MRI). This section presents the gradient - based image enhancement method for brain MR images which is based on the first derivative and local statistics. The preprocessing and enhancement method consists of two steps; first the removal of film artifacts such as labels and X - ray marks are removed from the MRI using tracking algorithm. Second, the removal of high frequency components using weighted median filtering technique [12]. It gives high resolution MRI compared to median filter, adaptive filter and spatial filter. The performance of the proposed method is also evaluated by means of peak single - to noise - ratio (PSNR), Average Signal - to - Noise Ratio (ASNR).

2.2 Segmentation

Image segmentation is the primary step and the most critical tasks of image analysis. Its purpose is that of extracting from an image by means of image segmentation. The mechanization of medical image segmentation has established wide application in diverse areas such as verdict for patients, treatment management planning, and computer - integrated surgery. Following are the segmentation algorithms that have been implemented:

2.2.1 Canny Method:

Edge detection is the approach used most frequently for segmenting images based on abrupt change in intensity. The canny edge operator works in a multi stage process. Canny algorithm was the only procedure capable of yielding a totally unbroken edge for the posterior boundary of the brain [5].

2.2.2 Otsu Method:

Otsu's thresholding is a non - linear operation that converts a grayscale image into a binary image where the two levels are assigned to pixel those that are below or above the specified threshold value [4]. The two levels in a binary image are assigned to pixels below or above the particular threshold. It is based on threshold range by statistical calculations. Otsu suggested minimizing the weighted sum of within - class variances of the object and background pixels to establish an optimum threshold [5].

2.2.3 Particle Swarm Optimization:

The algorithm based on swarm intelligence has been developed by adapting the collective behavior which is shown for searching food sources [6]. Each solution in PSO algorithm is a bird in the search space and it is called as a "particle". All particles have a fitness value evaluated by a fitness function and a velocity data that orients their fights. In the problem space, the particles move by following the existing most favorable solutions. PSO algorithm starts with a group of random generated solutions (particles) and optimal solution is investigated iteratively [7]. In each iteration, all particles are updated according to two best values. The first of these best values is that a particle found so far and is called “pbest”. The other one is the best value found so far by any particles in the population. This value is the global best value for the population and called as “gbest”

2.2.4 Watershed:

Watershed segmentation is a gradient - based segmentation technique. It considers the gradient map of the image as a relief map [9]. It segments the image as a dam. The segmented regions are called catchment basins. Watershed segmentation solves a variety of image segmentation problem. It is suitable for the images that have higher intensity value. To control over segmentation, marker controlled watershed segmentation is used. Sobel operator is suitable for edge detection [10]. In marker controlled watershed segmentation, Sobel operator is used to distinct the edge of the object [11].

2.3 Feature Extraction

Feature extraction is process of extracting quantitative information from an image such as color features, texture, shape and contrast. Here, we have used discrete wavelet transform (DWT) for extracting wavelet coefficients and gray-level co-occurrence matrix (GLCM) for statistical feature extraction.

2.3.1 Discrete Wavelet Transform:

The wavelet was used to analyze different frequencies of an image using different scales. Here, we are using discrete wavelet transform (DWT) which is a powerful tool for feature extraction. It was used to extract coefficient of wavelets from brain MR images. The wavelet localizes frequency information of signal function which was important for classification. By using 2D discrete wavelet transform, the images were decomposed into spatial frequency components which were extracted from LL (low–low) sub-bands and since HL(high–low) sub-bands have higher performance when compared to LL (low–low), we have used both LL (low–low) and HL (high–low) for better analysis which describes image text features [21].

2.3.2 Principal Component Analysis:

The most successful techniques that have been used in image recognition and compression is the Principal Component Analysis (PCA) and it is used to reduce the large dimensionality of the data. The basic approach is to compute the Eigen vectors of the covariance matrix of the original data and approximate it by a linear combination of the leading eigenvectors. By using PCA procedure, the test image can be identified by first, projecting the image onto the Eigen space to obtain the corresponding set of weights, and then comparing with the set of weights of the faces in the training set [21].

2.4 Classification

In classification tasks, there are often several candidate feature extraction methods available. The most suitable method can be chosen by training neural networks to perform the required classification task using different input features (derived using different methods). The error in the neural network response to test examples provides an indication of the suitability of the corresponding input features (and thus method used to derive them) to the considered classification task. Following are the classification algorithms that have been implemented:

2.4.1 Lazy IBK:

Lazy learners store the training instances until a query is sent to the system. The system generalize the training data pre receiving queries. Lazy learning has some advantages and disadvantages. The most important advantage is that the target function is locally approximated same as in K - nearest neighbor algorithm. This will enable the lazy system to work in parallel to solve multiple problems and handling any changes in the problem field at the same time [17]. On the other hand the disadvantages with using lazy learning represented by the large storage space requirement to store the whole training dataset.

IBK is a k- nearest neighbor classifier. A kind of different search algorithms can be used to speed up the task of finding the nearest neighbors. The distance function used with IBK is a parameter of the search method. The classifier keeps a limited number of training instances which is controlled by the window size option [18].

2.4.2 Support Vector Machine:

SVM is one of the classification technique applied on different fields such as face recognition, text categorization, cancer diagnosis, glaucoma diagnosis, microarray gene expression data analysis [15]. SVM utilizes binary classification of brain MR image as normal or tumor affected. SVM divides the given data into decision surface, (i.e. a hyper-plane) which divides the data into two classes. The prime objective of SVM is to maximize the margins between two classes of the hyper-plane [16]. Dimensionality reduction and precise feature set given as input to the SVM on the duration of training part as well as during the testing part. SVM is based on binary classifier which employs supervised learning to provide better results.

2.4.3 Convolutional Neural Networks:

Convolutional Neural Networks (CNNs) have proven to be very successful frameworks for image recognition. In the past few years, variants of CNN models achieve increasingly better performance for object classification [1].

Chapter 3. Project Design

This chapter contains a fully developed Software Project Management Plan for the project. The plan highlights the deliverables roles tasks and schedule for the project.

Our study deals with automated brain tumor segmentation and classification. Normally the anatomy of the brain can be viewed by MRI scan for diagnosis. Brain tumors usually cause unpredictable neurological harm to the body making detection of the tumor crucial for its treatment. It is important to predict the tumor and classify it so that appropriate treatment can be planned at an early stage. Different types of algorithms were developed for brain tumor detection but they have some drawback in effective detection and extraction of tumor.

3.1 Feasibility Analysis

Economic feasibility: Whether the firm can afford to build the software, whether its benefits should substantially exceed its cost. Our project is economically feasible. Our system uses academic version of MATLAB R2017a, which was very feasible, economically since it can be viewed as a one time investment.

Technical feasibility: Whether the technology needed for the system exists, how difficult it is to build. Our project is technically versatile system which can work on most platforms making it technically feasible to build requiring only few specifications. Software used for the project implementation is MATLAB. Basic technical knowledge of operating MATLAB software along with the classification toolbox is required for the developers.

Schedule Feasibility: How much time is available to build the new system, when it can be built? The project is entirely build from scratch to completion in a span of eight- nine months.

Ecological Feasibility: Whether the system has an impact on its environment. There are no adverse effects on the environment.

Operational feasibility: The system is easy to use and user-friendly. All maintenance issues will be handled efficiently. System is adaptable to most environments. Hence our system is operationally feasible.

3.2 Lifecycle Model

Waterfall model is non-iterative design process where System requirements are known initially and final outcome is determined. It progresses steadily downwards through above given faces.

When to use waterfall model:

- This model is used only when the requirements are very well known, clear and fixed.
- Product definition is stable.
- Technology is understood.
- There are no ambiguous requirements
- Ample resources with required expertise are available freely

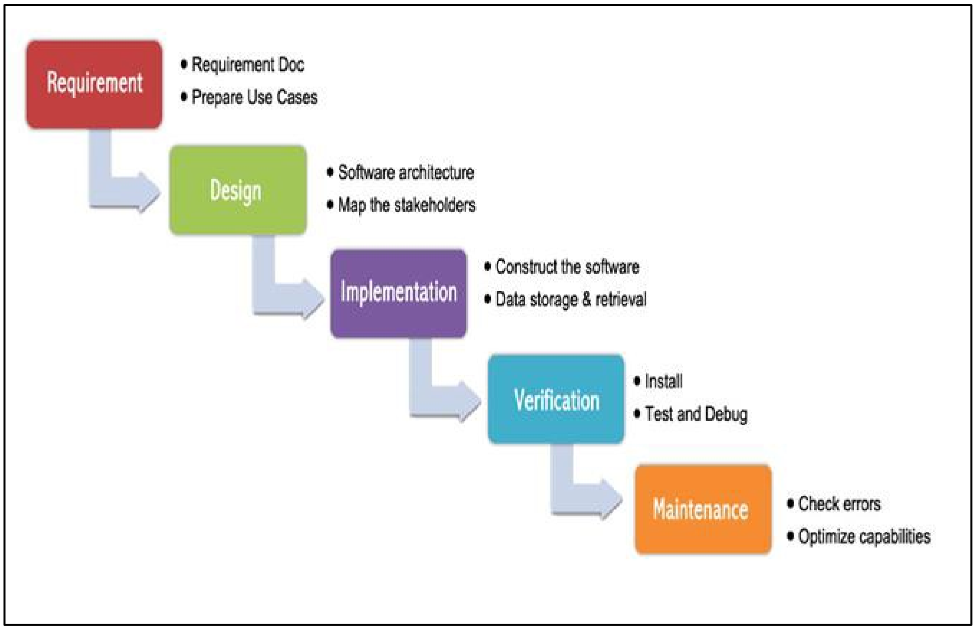


Figure 1: Waterfall Model

Functionality 1: Requirements Gathering

- MATLAB 2017a software and brain MRI scan datasets.
- At least 100 datasets of brain scans. The vaster a dataset, more accurate will the results be.
- Gathering information about different filters such as - Gaussian, Mean filters etc. for seeing which gives the best result for our study
- Clustering algorithms for segmentation- Canny, Otsu, Particle Swarm Optimization etc. Feature extraction techniques such as PCA and GLCM.

Functionality 2: Design

- Designing the process overview from applying filters to segmentation, feature extraction and classification of brain tumor.

Functionality 3: Implementation

- Implementing all algorithms in MATLAB.

Functionality 4: Verification

- Verifying it by testing it on minimum 50 datasets of scans.

Functionality 5: Maintenance·

- Maintaining from time to time for its efficiency.

Project deliverables:

- Software Project Management Plan
- Software Requirements specifications
- Software Design Description
- System Test Document
- User Interface Module
- Final Product

3.3 Project Cost and Time Estimation

Project only used the academic version of the MATLAB R2017. No other additional cost was incurred. The system will be operational by 15th April, 2018.

3.4 Resource Plan

Resources:

There are four members working on this project – Apurva Mehta, Hitesh Jaiswal, Shreyansh Kotak and Yash Pasar. Each of the team members will be involved in almost all the activities and will play a responsible role in the development of the prototype. The model being complex, it will include participation of every member of the team.

The project is guided by Prof. Poonam Bhogle who provided necessary assistance and guided all the team members. She analyzed all the documents and reviewed them by submitting her opinions which were taken into due consideration.

Activities:

1. Write and Submit Proposal
2. Conduct literature survey
3. Data set collection.
4. Segmentation, feature extraction and Classification algorithms
5. Accuracy testing.
6. Write Software requirement specification (SRS)
7. Write Software project management plan
8. Write Software design description
9. Extensive accuracy testing
10. Modifications
11. Develop GUI

Note: All values in Table 1 below are in percentage (%).

	Activity/ Resource	1	2	3	4	5	6	7	8	9	10	11
April	A	25										
	H	15										
	S	30										
	Y	30										
May	A		30									
	H		10									
	S		15									
	Y		25									
June	A			30								
	H		30	10								
	S			15								
	Y			25								
July	A		20	10	15							
	H				30							
	S				30							
	Y		20	20	15							
Aug.	A											
	H					55						
	S					45						
	Y											
Sept.	A						25					
	H						25					
	S						25					
	Y						25					
Oct.	A							30				
	H							15				

	S							15				
	Y							30				
Nov.	**A**								25			
	H								15			
	S								30			
	Y								30			
Dec.	**A**								15	25		
	H									30		
	S									15		
	Y									20		
Jan.	**A**										10	
	H									10	45	
	S										35	
	Y										10	
Feb.	**A**									20	30	
	H										15	
	S										20	
	Y										35	
March	**A**									50		
	H										40	
	S										20	
	Y										20	
April	**A**											25
	H											25
	S											25
	Y											25

Table 1: Activities conducted by resources every month

3.5 Task & Responsibility Assignment Matrix

Person	**Responsibility**	**Task**
Apurva Mehta	• Data Gathering • Implementation • Training • Documentation • Testing	• Study of various IEEE papers, research on the previous projects and implementation techniques to be used. • Data Collection and Analysis. • Write Software requirement specification(SRS) • Developing UML diagrams. • Implementation of Functionalities for processing, and portfolio management. • Generating timely reports and activities throughout our project

Hitesh Jaiswal	• Data Gathering • Implementation • Training • Documentation • Testing	• Study of various IEEE papers, research on the previous projects and implementation techniques to be used. • Data Collection and Analysis. • Write and submit proposal • Conduct literature survey • Designing GUI through which users will interact, designing simple functional model of project. • Implementation of Functionalities for processing, and portfolio management. • Testing each and every module for Debugging bugs and errors. • Generating timely reports and activities throughout our project
Shreyansh Kotak	• Implementation • Training • Testing	• Study of various IEEE papers, research on the previous projects and implementation techniques to be used. • Designing GUI through which users will interact, designing simple functional model of project. • Implementation of Functionalities for processing, and portfolio management. • Testing each and every module for Debugging bugs and errors • Generating timely reports and activities throughout our project

Yash Pasar	• Implementation • Training • Testing	• Study of various IEEE papers, research on the previous projects and implementation techniques to be used. • Designing GUI through which users will interact, designing simple functional model of project. • Implementation of Functionalities for processing, and portfolio management. • Testing each and every module for Debugging bugs and errors • Generating timely reports on activities.

Table 2: Task & Responsibility Assignment Matrix

3.6 Project Timeline Chart

Task	Start Date	End Date	Duration
Proposal	3rd March	15th March	12 days
Literature Survey	10th March	9th June	89 days
Otsu, Canny, PSO and Watershed Algorithms	9th July	6th October	89 days
GLCM Features (PCA+DWT)	16th October	26th October	20 days
CNN	30th October	20th January	50 days
SVM and Lazy IBK	30th January	10th February	11 days
Extensive accuracy testing	1st February	15th February	14 days
SRS	4th March	8th March	4 days
Software Project Management Plan	24th December	30th December	6 days
Software Design Description	2nd February	8th February	6 days
Report	9th December	12th December	3 days
Extensive accuracy testing on more brain scans.	15th February	15th March	30 days
Modifications	16th March	13th April	27 days
GUI	30th January	10th February	45 days

Table 3: Project Timeline Chart

Gantt Chart:

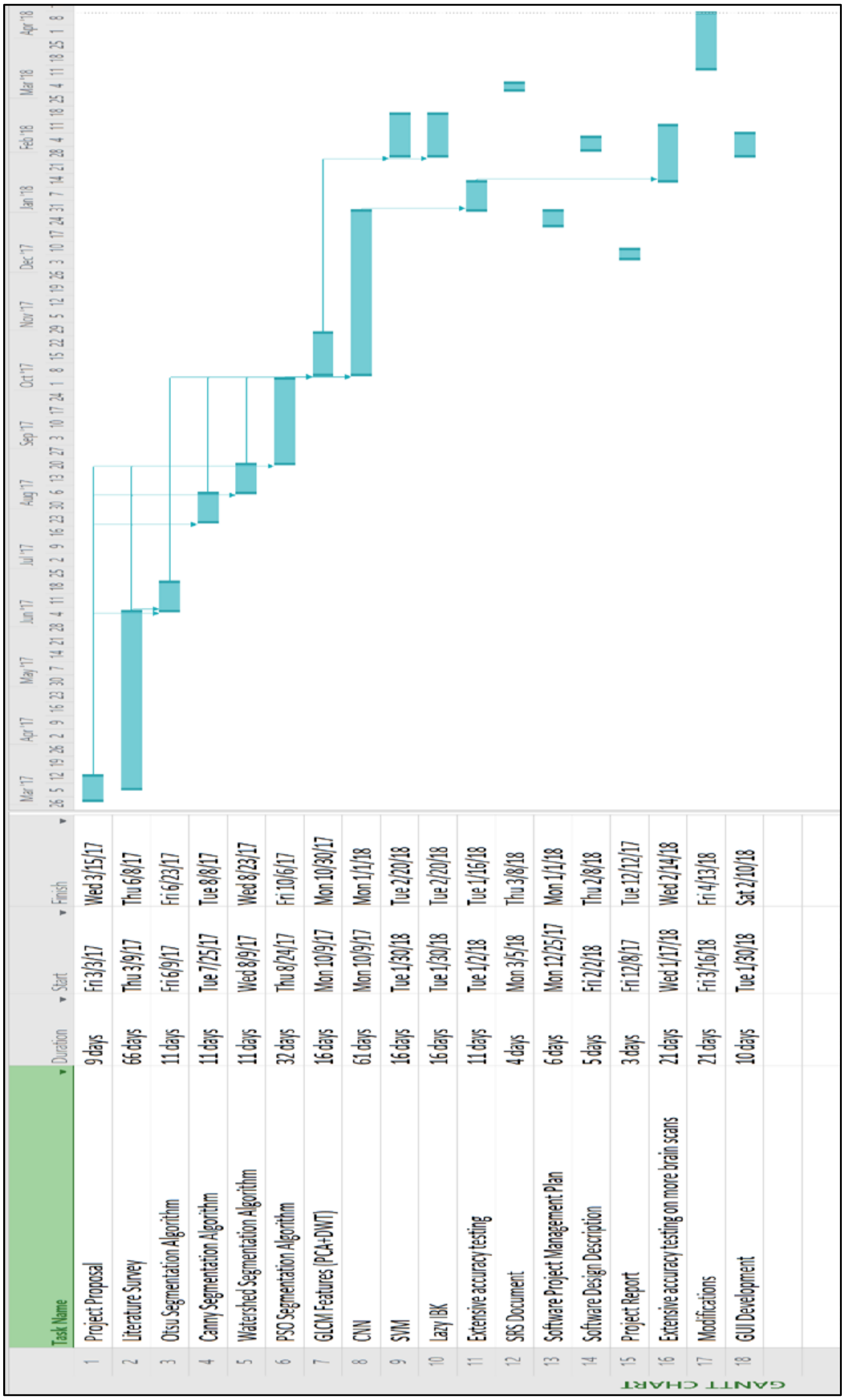


Figure 2: Gantt Chart

3.7 Software Architecture Diagram

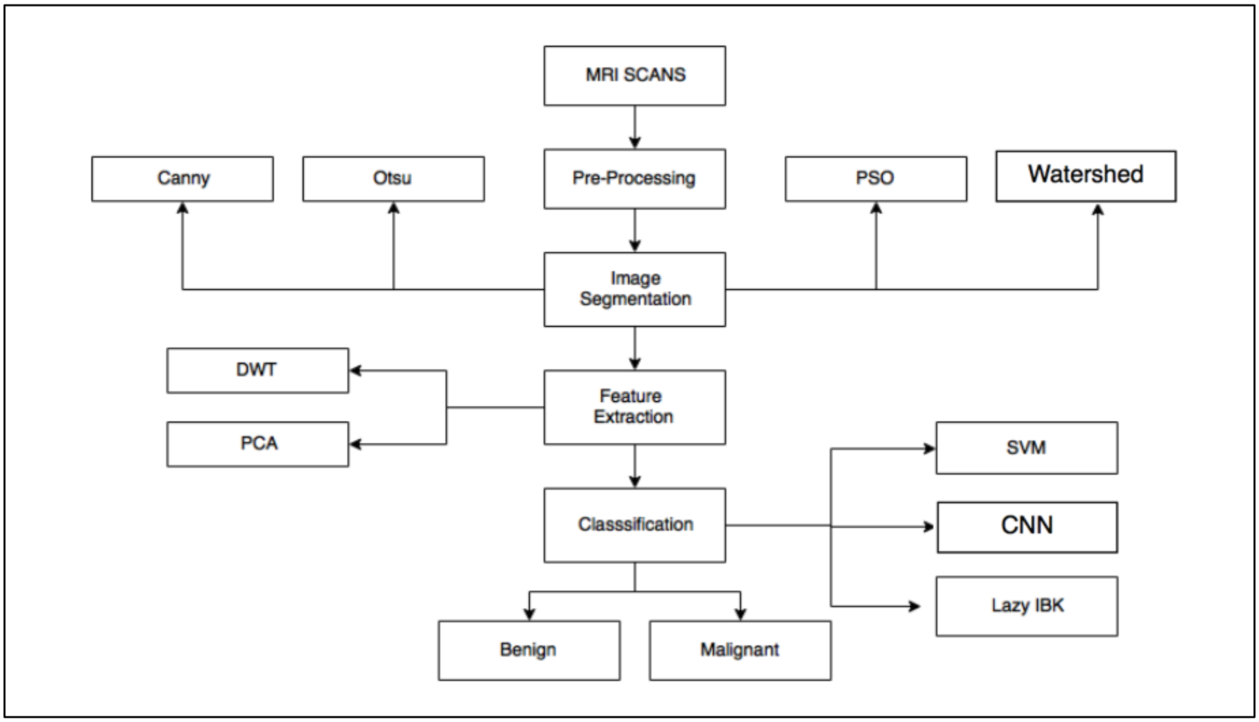


Figure 3: Software Architecture Diagram

3.8 Architectural Style and Justification

The diagram (Figure 3) shows the various stages in the development of our system. The diagram shows interaction between the various components of the application and their position in the development hierarchy.

This style hence, is appropriate for the selected problem because all the modules in the selected problem function independently. The communication is strictly through message passing connectors. The flow of the system is from the top to the bottom.

3.9 Software Requirements Specification Document

Introduction:

1. Product Overview

Our objective is to develop a system incorporating image processing, and computer vision techniques for enhancement, segmentation of breast tumors. Our study aims at enhancing the current accuracy (diagnostic) of digital MRI scans using industry standard simulation software tool, MATLAB and the online dataset.

These techniques involve pre-processing of digital MRI scans by resizing them and then apply the proposed algorithms for segmentation, feature extraction and classification. The system is expected to improve the efficiency, sensitivity and specificity of brain tumor screening, and possibly reduce health care costs by decreasing the need for follow-up procedures such as biopsy.

2. Specific Requirements

2.1 User Interfaces

- The end users of our system will be neurosurgeons.
- The process time delay between selecting a Tumor image and producing an output when using a user interface should be minimal.

2.2 Software product features

- The software is aimed to be simplistic, with minimal complexity and provides ease of use for even beginner users. Minimal training, if at all will be required to use the software product being built, which is provided by referring the user manuals and following the GUI diagrams and familiarizing them with it.

- Rigorous training as well as testing of the dataset by a technical expert, in order to meet the required deliverables and have maximum throughput and efficiency for our system.
- Automation to ease the user of manual tasks is the primary aim of the project.

2.3 Software system attributes

- Reliability:
  - Optimal functionality of software for as long as it is installed on the device.
  - Minimal maintenance of system that can be automated as well.
  - Must be adaptable to changing hardware and operating environment, making it versatile over time.
  - End user should be able to completely rely on the efficiency of the software.
- Availability:
  - The software should be available on any platform to the user whenever required.
  - In case of failure of software, consistency factor must be preserved
- Portability:
  - The software should be able to be run on 64 bit Windows operating system.
  - It should be able to work seamlessly on any version of Windows from Windows 7 onwards on system and hardware any operating providing necessary portability
- Maintainability:
  - Time to time maintenance of software, along with necessary updates is of utmost importance. This is especially needed if or when the error between predicted tumor and the actual tumor tends to vary much which may affect the efficiency of product. In such cases, the dataset is used for classification algorithms in the software to train to achieve certain level of accuracy.

- Performance:
  - A good data storage repository having large capacity will be required to store the software as well as the MRI scan datasets stored as input, as well as data collected from the inferences made as final output.
- Security:
  - Administrator of the software must be given security so that the internal structure of the software stays unaltered by any non-authorized entity which may cause potential harm to the system.
  - Access rights must be clearly and explicitly specified and practiced lawfully.
  - Any kind of line-crossing by unauthorized personnel must be immediately restricted for minimal security damage.

2.4 Database Requirements

The database will contain the predicted data and the actual perceived data, as a percept sequence. The database consist of MRI brain scans taken as input data, that are loaded to the system for training and classification, which are updated itself every time the patient database increases. Only the physician and neurosurgeons will have access rights to modify this database so that sensitive information does not leak.

2.5 Performance Requirements

To avoid any latency in operation or lag in performance of the application system, it is necessary to ensure processor is not slow, or that an outdated version of MATLAB is under use currently.

2.6 Safety Requirements

- User should not share dataset of patient with others.
- Only the physician and neurosurgeons will have access rights to modify this database so that sensitive information does not leak.
- User should select the region of affected brain out of the MRI scans to provide for accurate mapping
- User should answer executive call when needed.

2.7 Security Requirements

- User and administrator should not share dataset of patient with another user.
- Only the physician and neurosurgeons will have access rights to modify this database so that sensitive information does not leak.

2.8 Hardware and Software Platform requirements

Software Requirements:

- MATLAB 2017a
- Dataset obtained from pathology labs
- Online dataset from www.cancerimagingarchive.net
- Microsoft Office

Hardware Requirements:

- Intel Dual Core Dual Processor or advanced version
- Minimum 8GB of RAM
- Minimum 1 GB of Hard disk Space

3.10 Software Design Document

3.10.1 Use Case Diagram

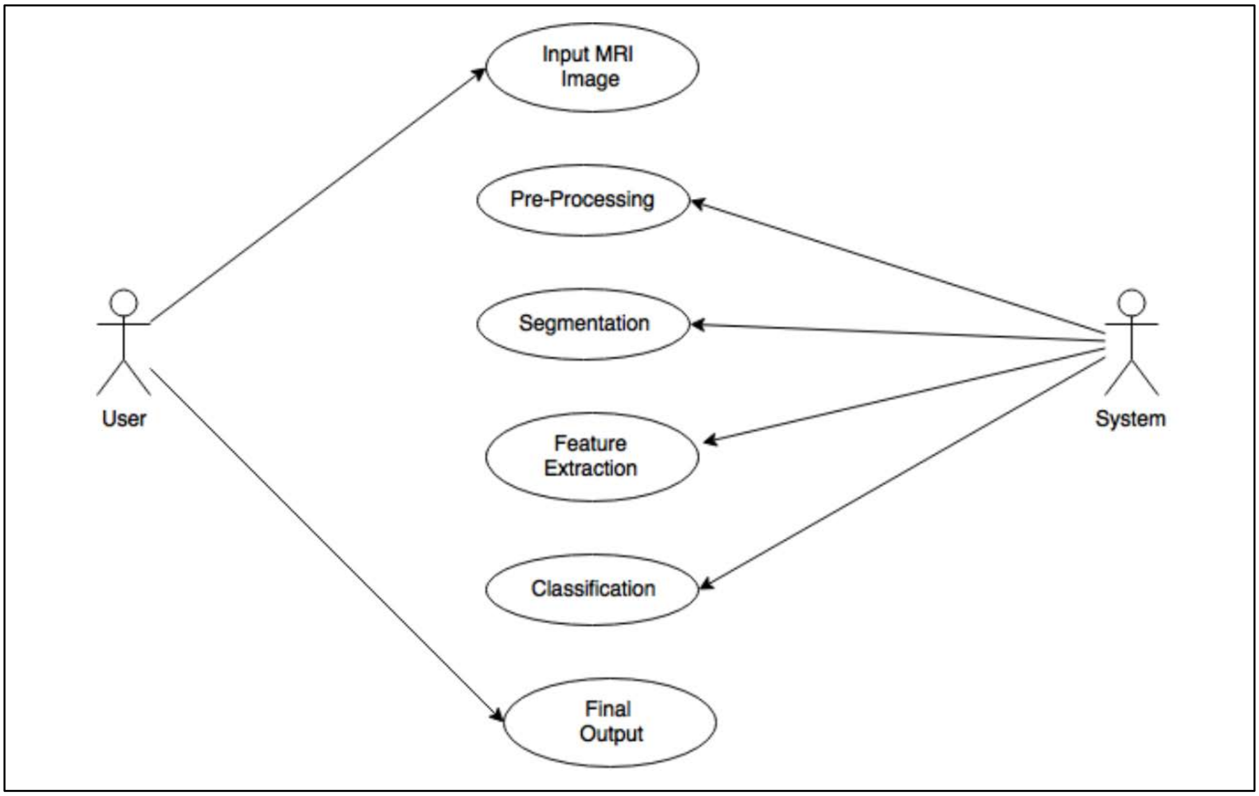


Figure 4: Use Case Diagram

- The use case diagram consists of two actors, who interact with the software.
- The User: The user takes in input image and sees the final output
- The System: The System performs all clustering, feature extraction, classification and training algorithms.

3.10.2 Class Diagram

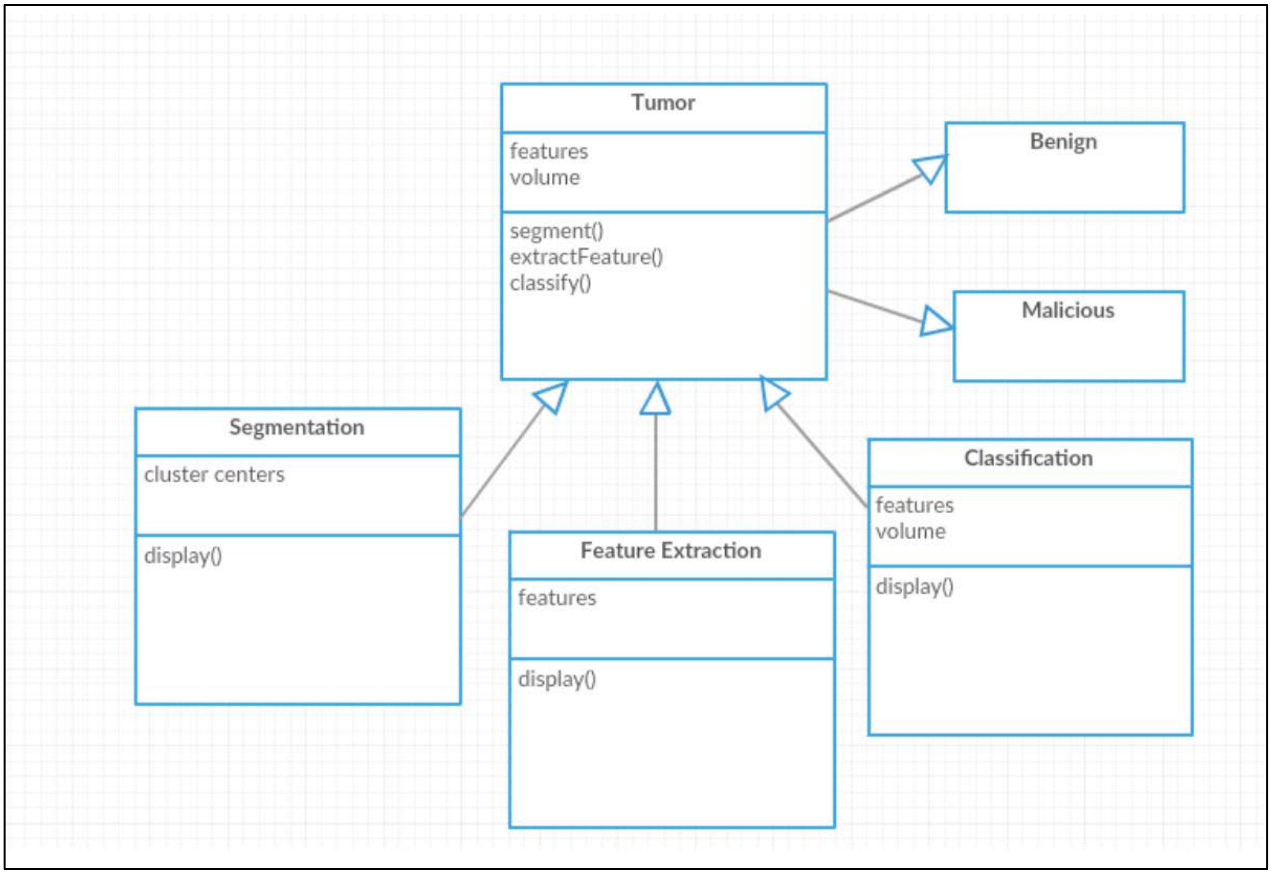


Figure 5: Class Diagram

3.10.3 User Interface:

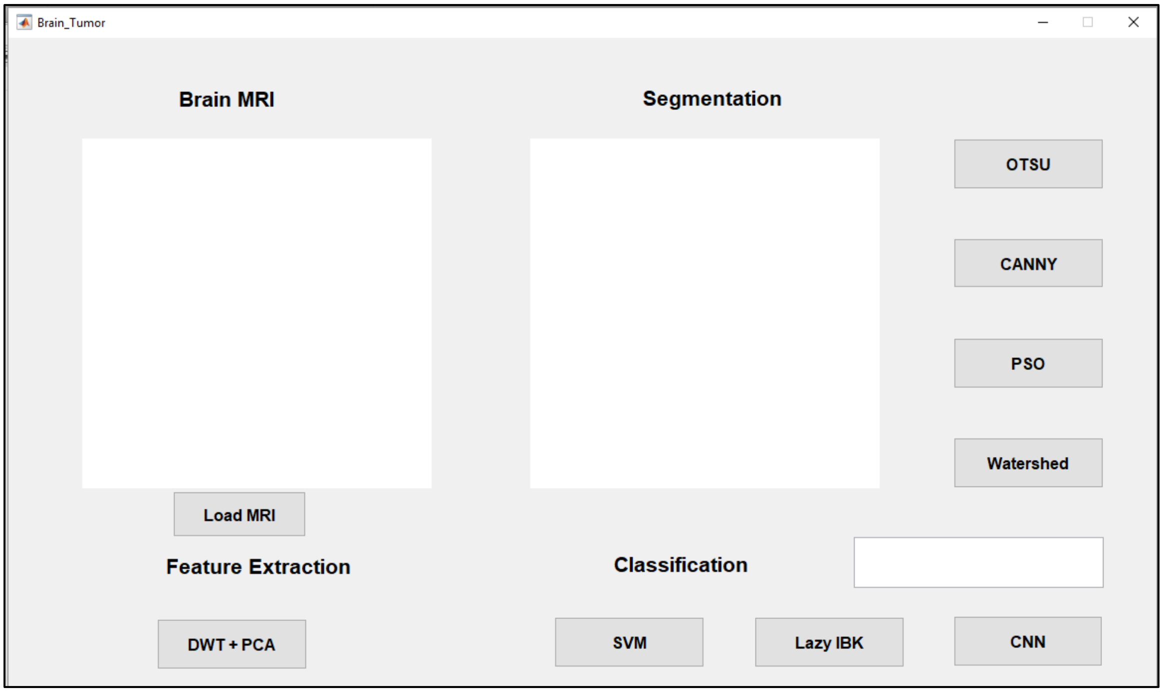


Figure 6: Graphical User Interface

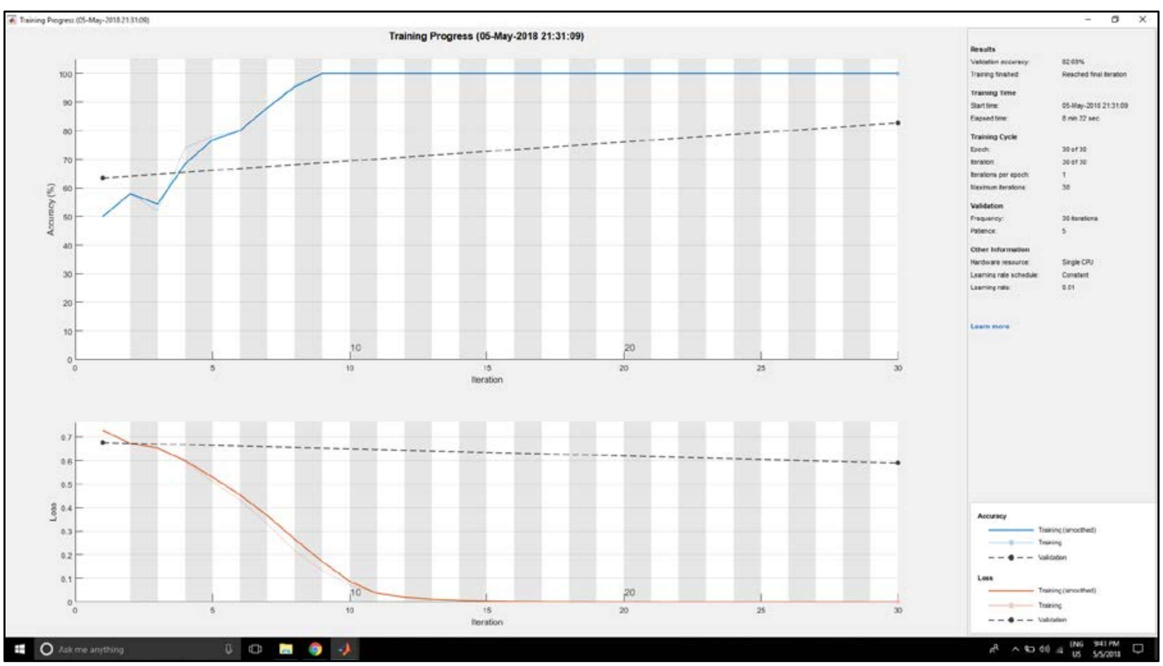


Figure 7: Training under Progress (OTSU – CNN)

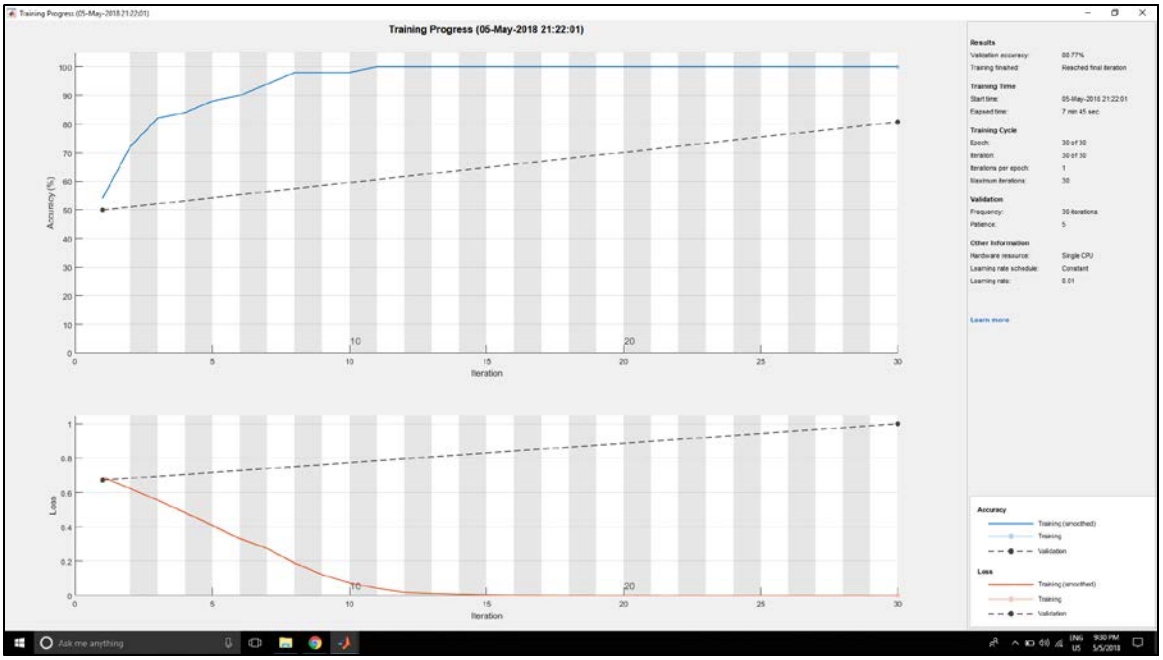


Figure 8: Training under Progress (Canny – CNN)

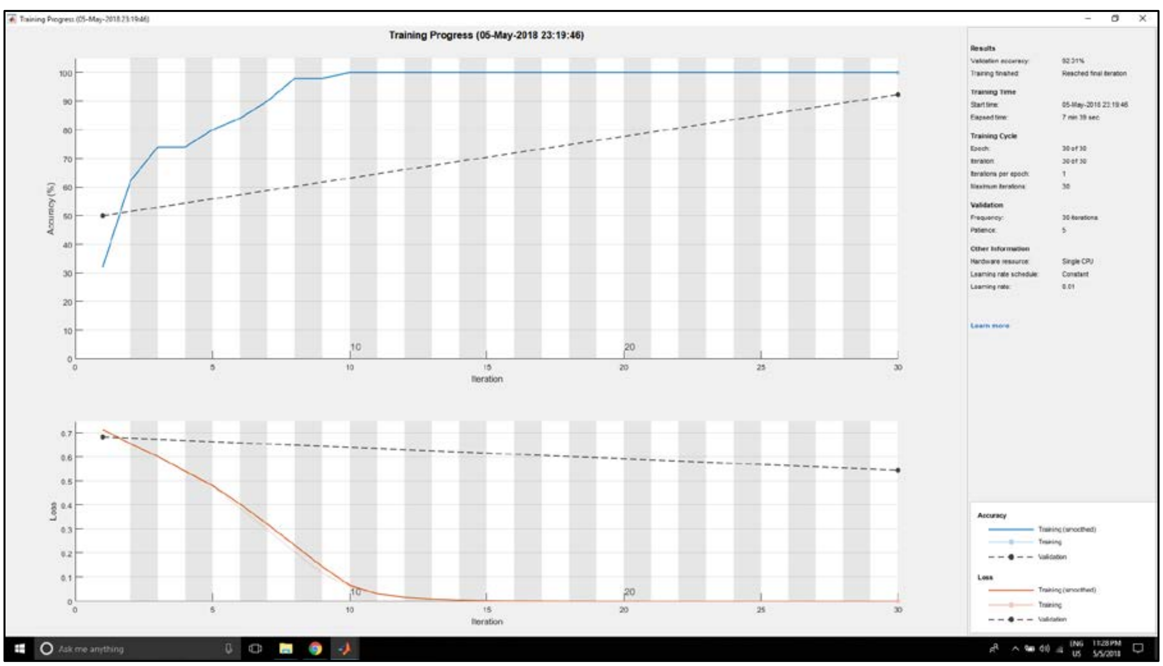


Figure 9: Training under Progress (PSO – CNN)

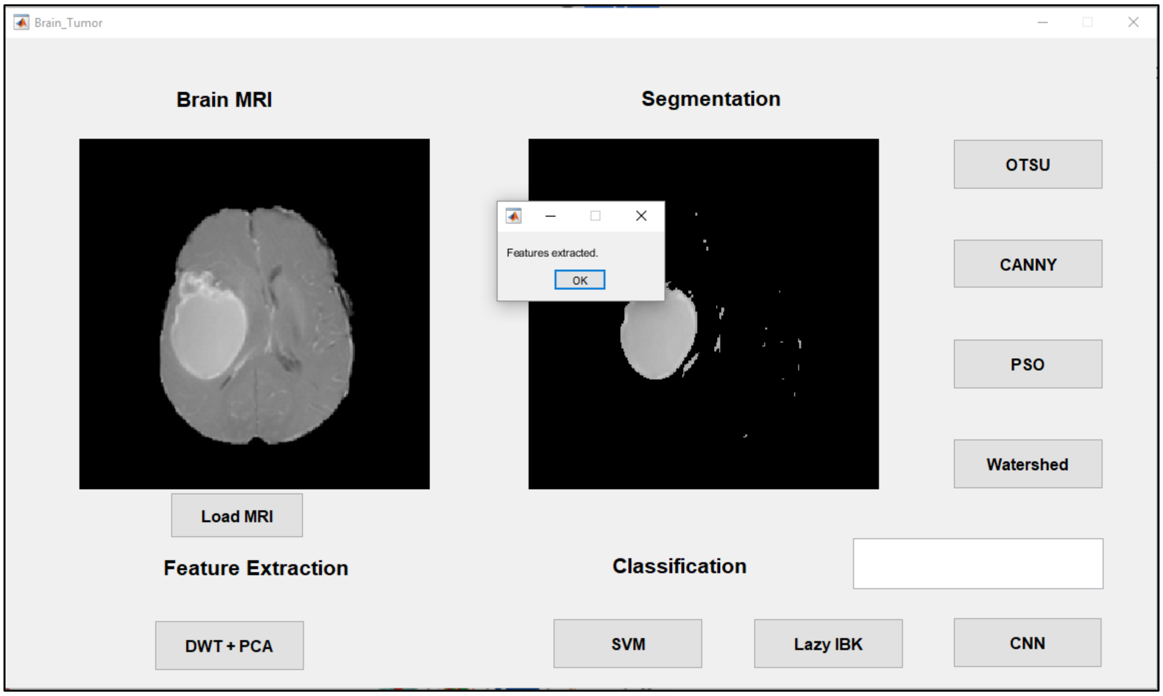


Figure 10: Features Extracted

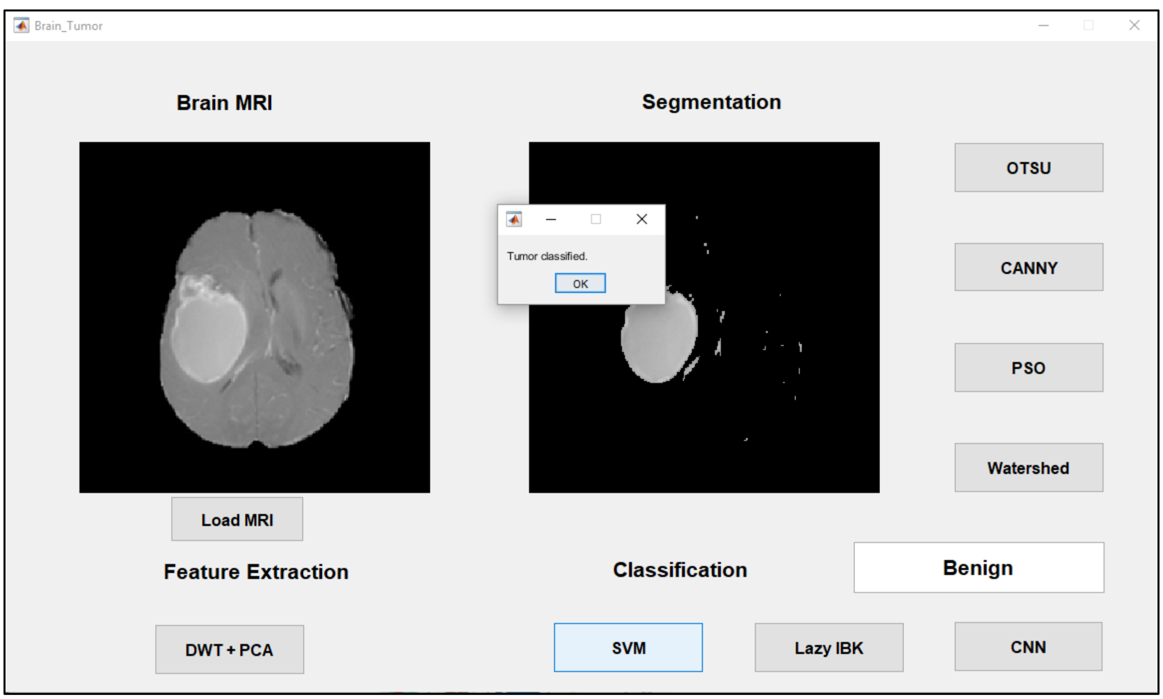


Figure 11: Classified Output (Benign)

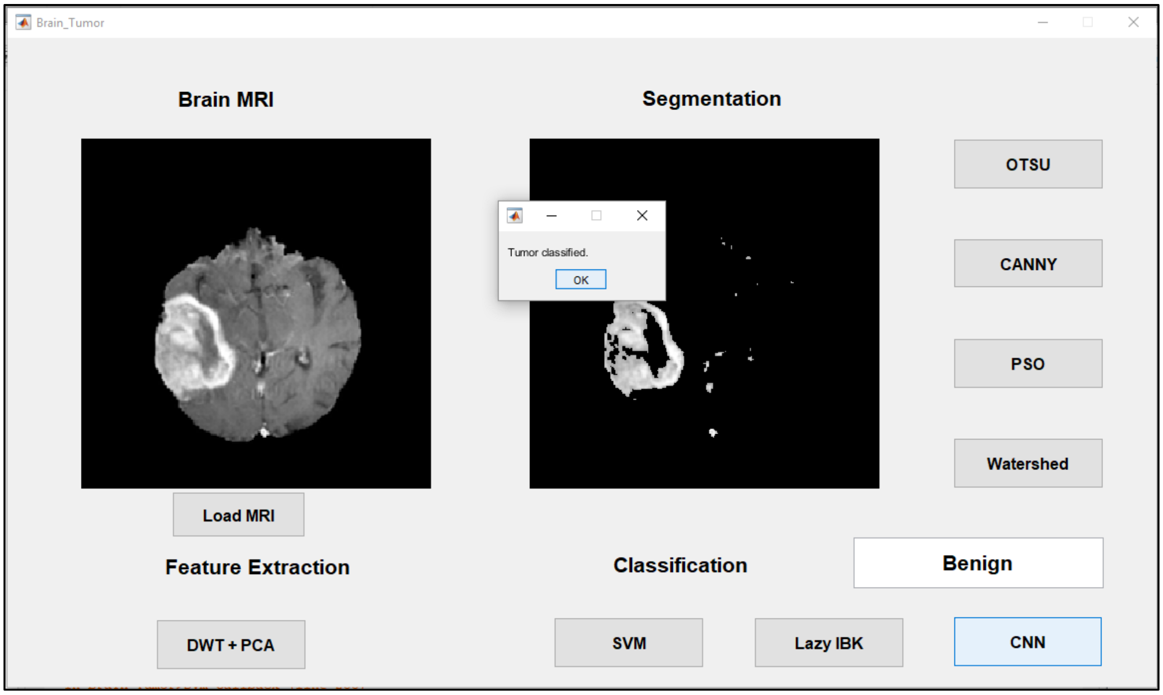


Figure 12: Classified Output (Benign)

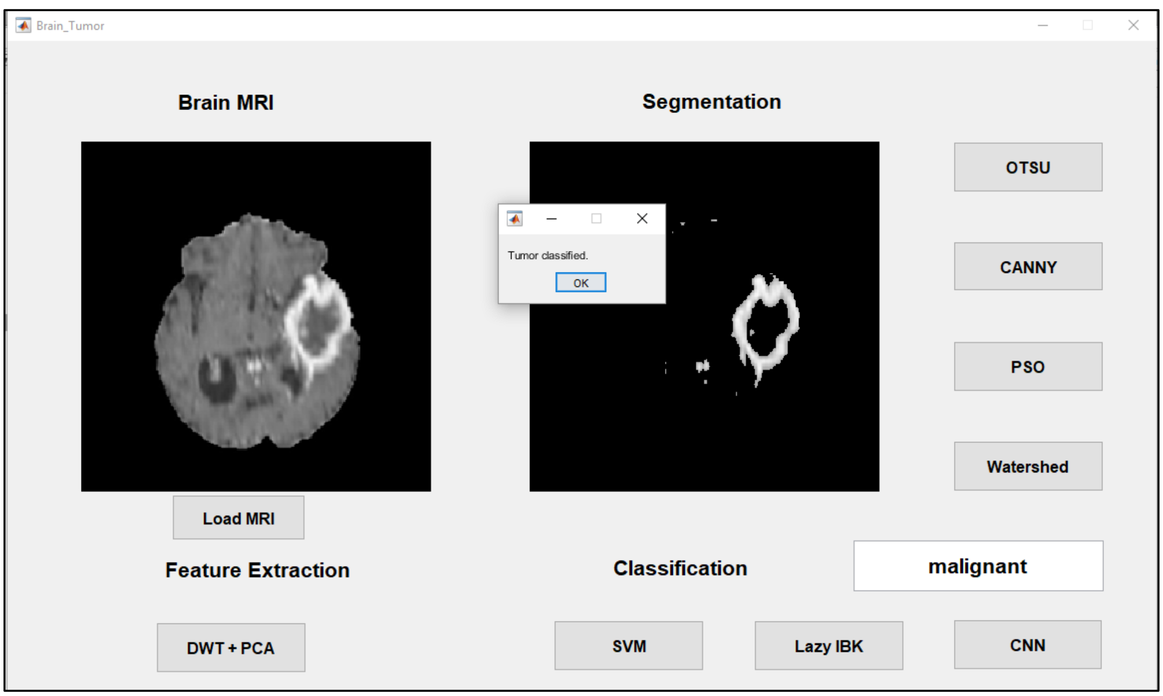


Figure 13: Classified Output (Malignant)

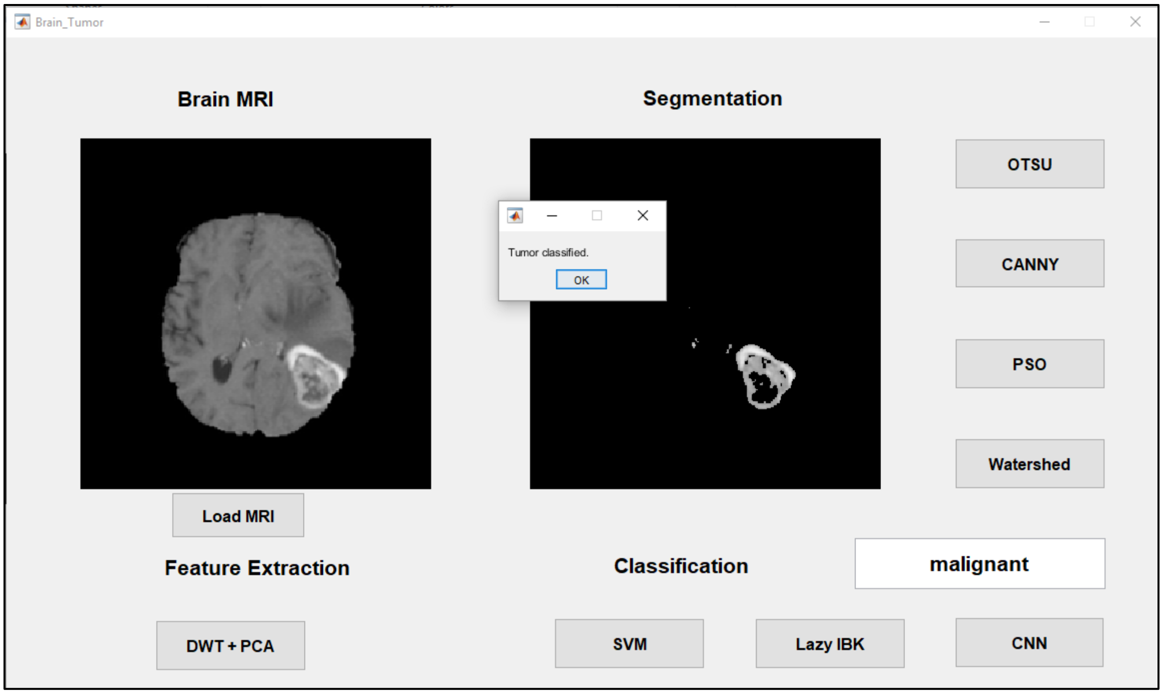


Figure 14: Classified Output (Malignant)

3.10.4 Component Diagram

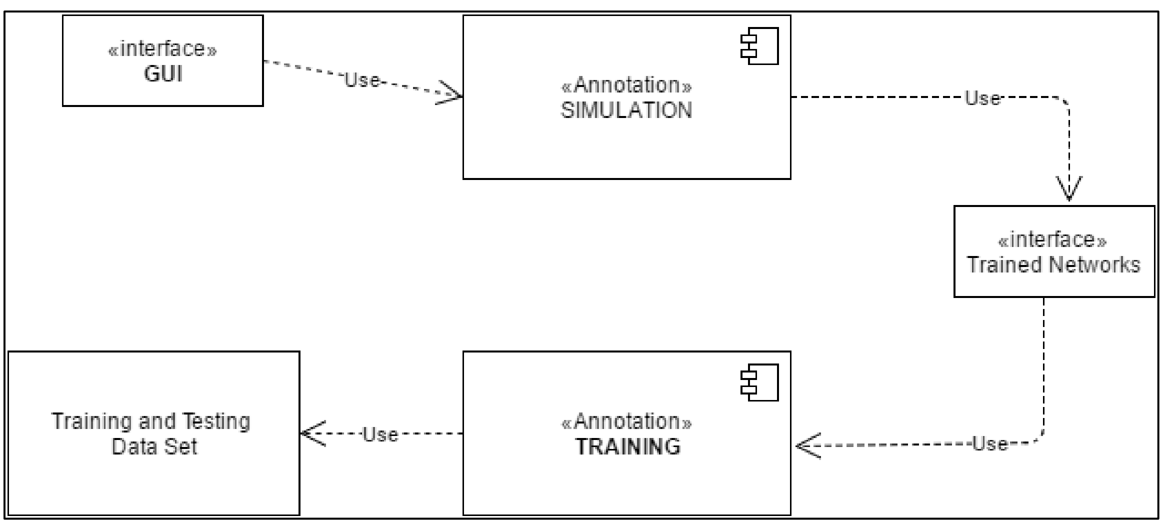


Figure 15: Component Diagram

3.10.5 Deployment Diagram

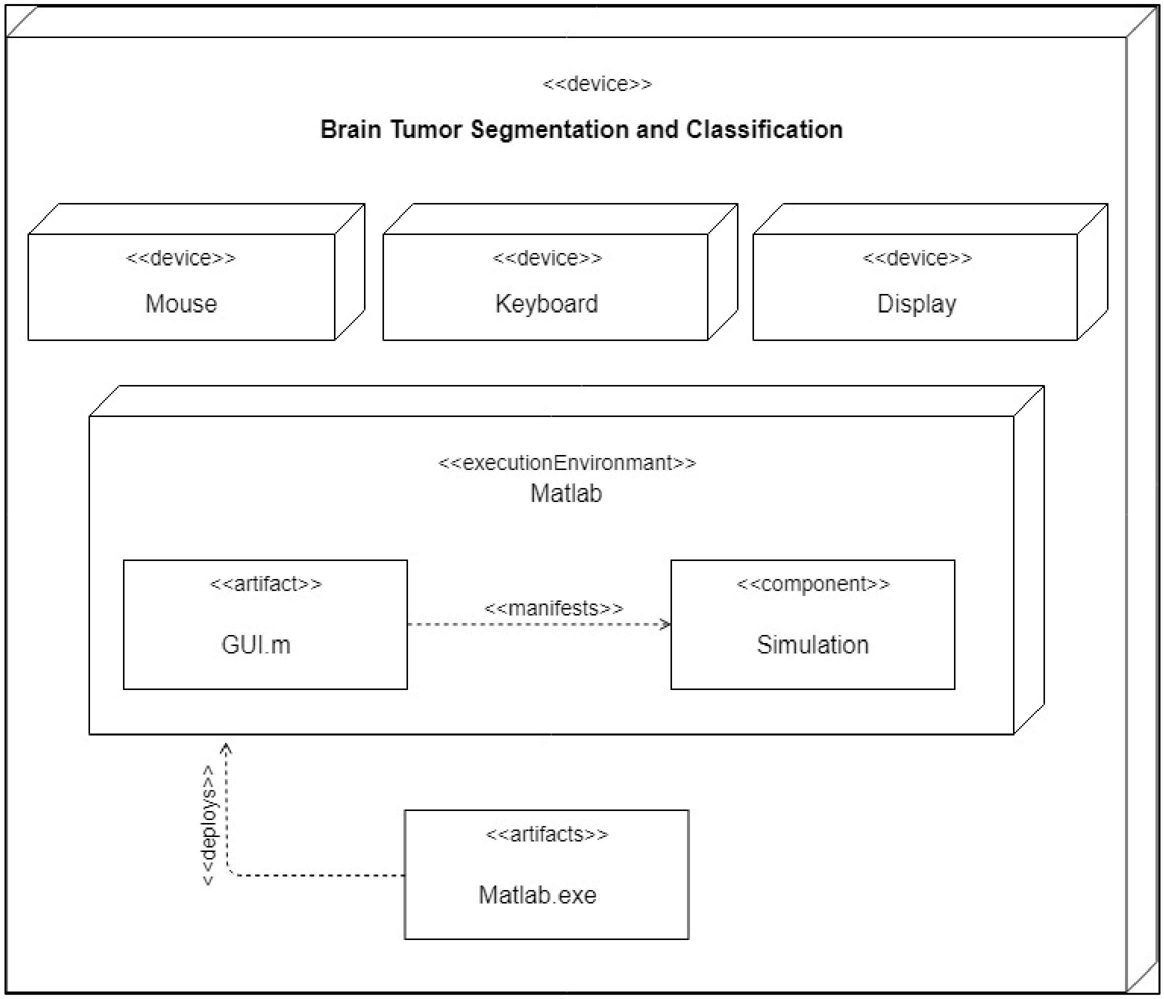


Figure 16: Deployment Diagram

Chapter 4. Project Implementation

This chapter discusses the implementation of the project – the various algorithms, testing approaches and the results.

4.1 Methodology

We have implemented nine algorithms in this project. A detailed explanation and the various outputs are shown below:

4.1.1 Dataset Used for Analysis

Data was collected from various verified sources and then segregated into two types:

- Cancerous (Malignant)
- Non-cancerous (Benign)

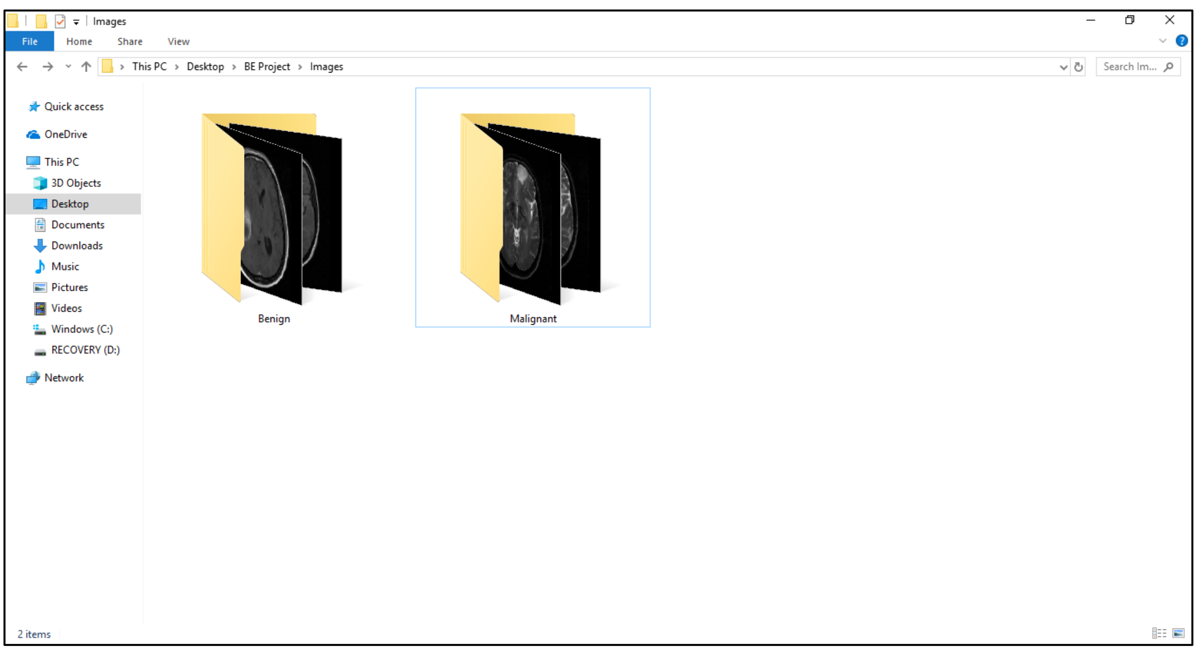


Figure 17: Segregated Dataset

4.1.2 Segmentation

4.1.2.1 Canny Algorithm

Canny edge detection is a technique to extract useful structural information from different vision objects and dramatically reduce the amount of data to be processed. It has been widely applied in various computer vision systems. Canny has found that the requirements for the application of edge detection on diverse vision systems are relatively similar. Thus, an edge detection solution to address these requirements can be implemented in a wide range of situations.

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

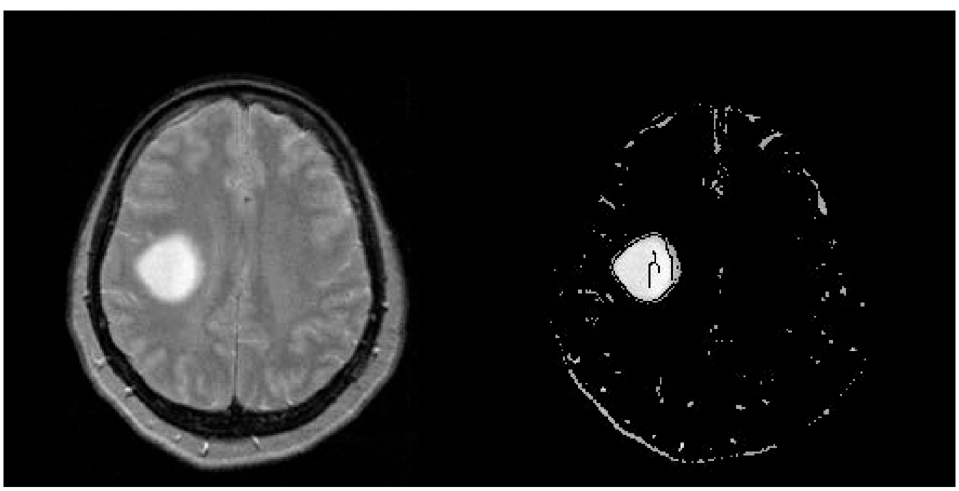


Figure 18: Output from Canny Algorithm

4.1.2.2 Otsu Algorithm:

In image processing, Otsu's method, named after Nobuyuki Otsu, is used to automatically perform clustering-based image thresholding, or, the reduction of a graylevel image to a binary image. The algorithm assumes that the image contains two classes of pixels following bi-modal histogram (foreground pixels and background pixels), it then calculates the optimum threshold separating the two classes so that their combined spread (intra-class variance) is minimal, or equivalently (because the sum of pairwise squared distances is constant), so that their inter-class variance is maximal. The algorithm proceeds as follows:

1. Compute histogram and probabilities of each intensity level
2. Set up initial class probabilities and mean.
3. Step through all possible thresholds t = 1,2 …….maximum intensity
    1. Update class probabilities and mean
    2. Compute inter class variance
4. Desired threshold corresponds to the maximum inter class variance.

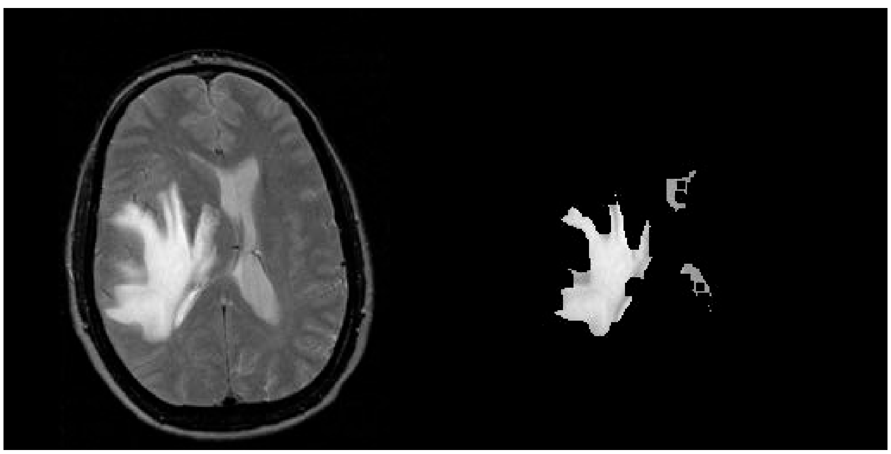


Figure 19: Output from Otsu Algorithm

4.1.2.3 Watershed Algorithm

Any grayscale image can be viewed as a topographic surface where high intensity denotes peaks and hills while low intensity denotes valleys. We start filling every isolated valleys (local minima) with different colored water (labels). As the water rises, depending on the peaks (gradients) nearby, water from different valleys, obviously with different colors will start to merge. To avoid that, we build barriers in the locations where water merges. We continue the work of filling water and building barriers until all the peaks are under water. Then the barriers created gives the segmentation result.

But this approach gives over-segmented result due to noise or any other irregularities in the image. Hence, we have implemented a marker-based watershed algorithm where you specify which are all valley points are to be merged and which are not. It is an interactive image segmentation. Label the region which we are sure of being the foreground or object with one color (or intensity), label the region which we are sure of being background or non-object with another color and finally the region which we are not sure of anything, label it with 0. That is the marker. Then we apply watershed algorithm. Then our marker will be updated with the labels we give, and boundaries of objects will have a value of -1.

Marker-controlled watershed segmentation follows this basic procedure:

1. Compute a segmentation function.
2. Compute foreground markers.
3. Compute background markers.
4. Modify the segmentation function so that it only has minima at the foreground and background marker locations.
5. Compute the watershed transform of the modified segmentation function.

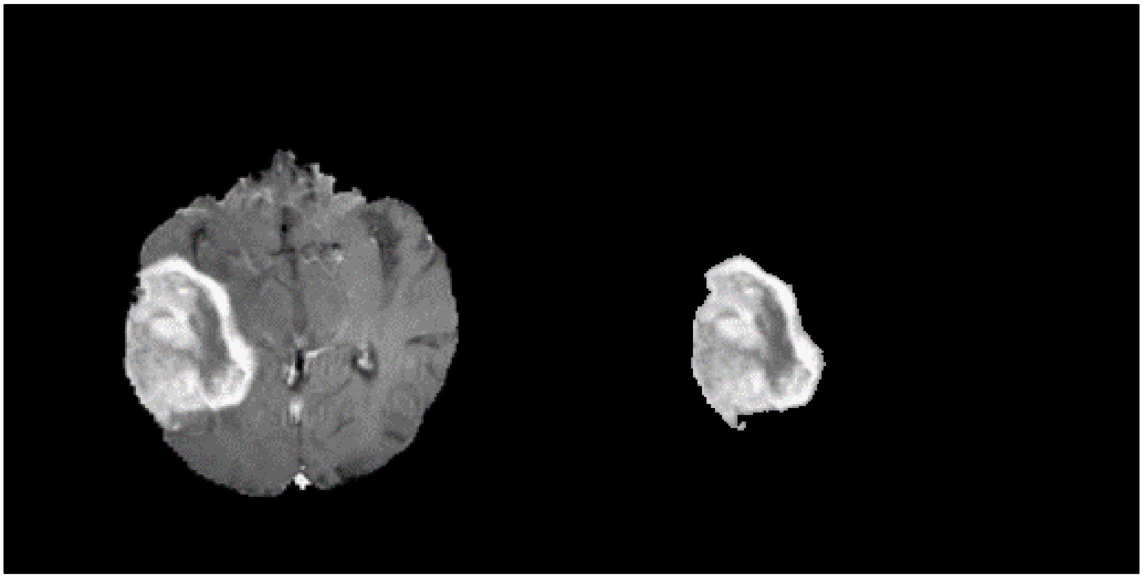


Figure 20: Output from Watershed Algorithm (Malignant)

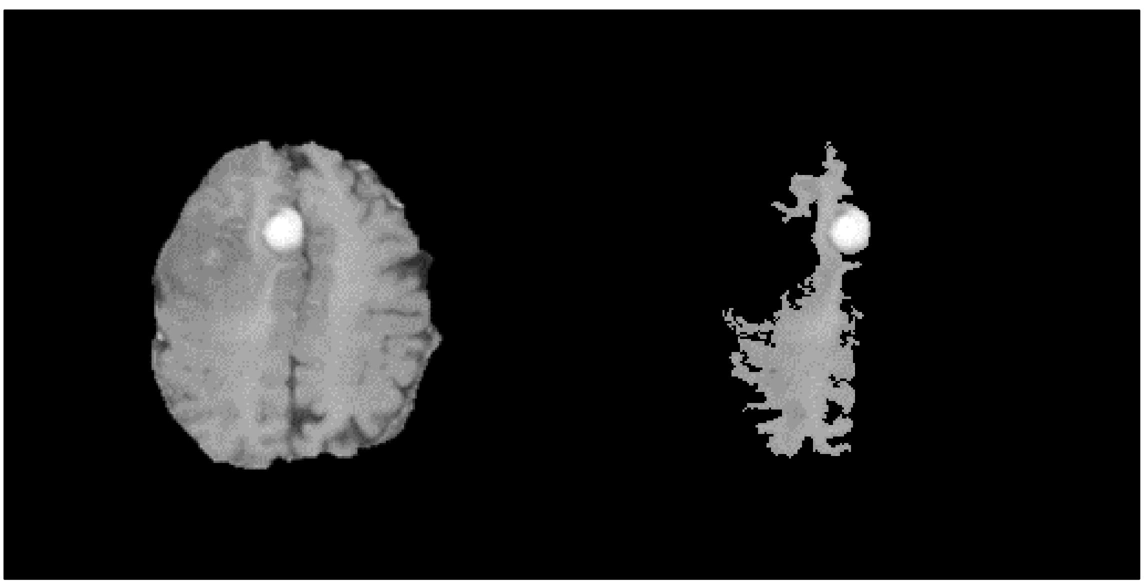


Figure 21: Output from Canny Algorithm (Benign)

4.1.2.4 Particle Swarm Optimization (PSO)

The PSO algorithm starts by generating random positions for the particles, within an initialization region Θ′⊆Θ. Velocities are usually initialized within Θ′ but they can also be initialized to zero or to small random values to prevent particles from leaving the search space during the first iterations. During the main loop of the algorithm, the velocities and positions of the the particles are iteratively updated until a stopping criterion is met. The update rules are:

$$\vec{v}_i^{\,t+1} = w\vec{v}_i^{\,t} + \varphi_1 \vec{U}_1^{\,t}(\vec{b}_i^{\,t} - \vec{x}_i^{\,t}) + \varphi_2 \vec{U}_2^{\,t}(\vec{l}_i^{\,t} - \vec{x}_i^{\,t}),$$

$$\vec{x}_i^{\,t+1} = \vec{x}_i^{\,t} + \vec{v}_i^{\,t+1},$$

where w is a parameter called inertia weight, φ1 and φ2 are two parameters called acceleration coefficients, U->t1 and U⃗ t2 are two n×n diagonal matrices in which the entries in the main diagonal are random numbers uniformly distributed in the interval [0,1] . At each iteration, these matrices are regenerated. Usually, vector l-> ti , referred to as the neighborhood best, is the best position ever found by any particle in the neighborhood of particle pi , that is, f(l⃗ ti)≤f(b⃗ tj)∀pj∈Ni. If the values of w ,φ1 and φ2 are properly chosen, it is guaranteed that the particles' velocities do not grow to infinity.

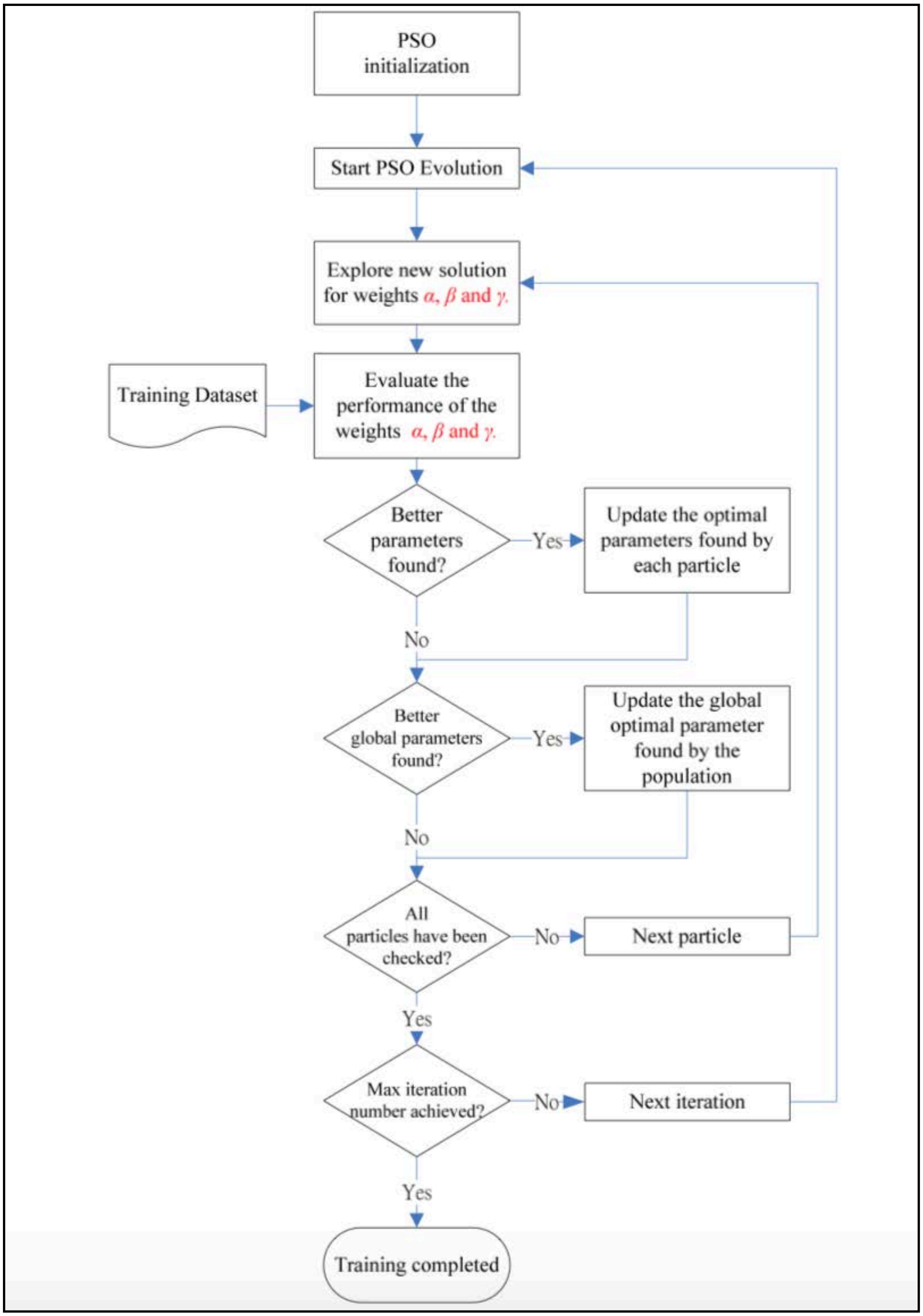


Figure 22: PSO Algorithm Flow Chart

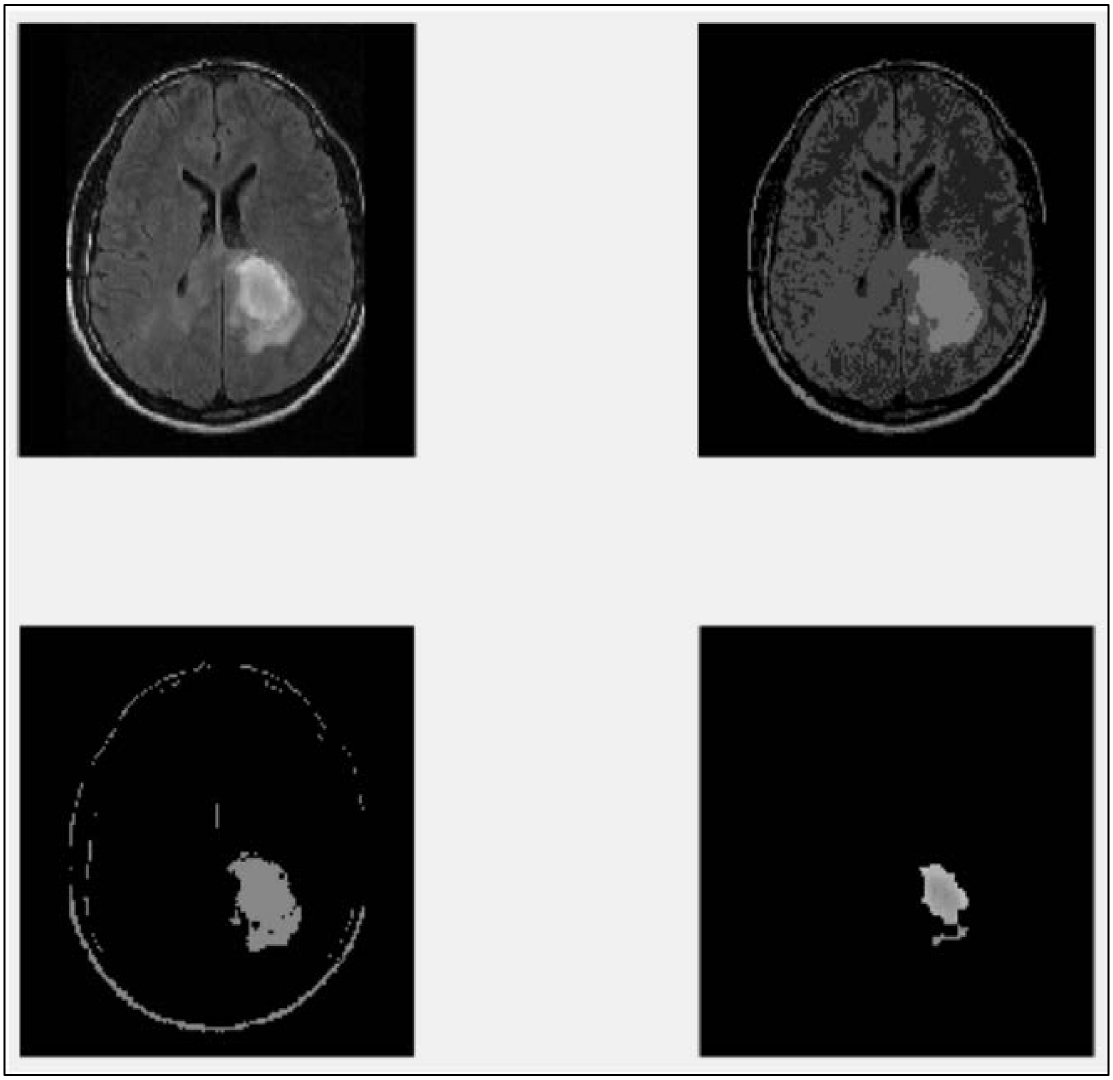


Figure 23: Output from PSO Algorithm (Malignant)

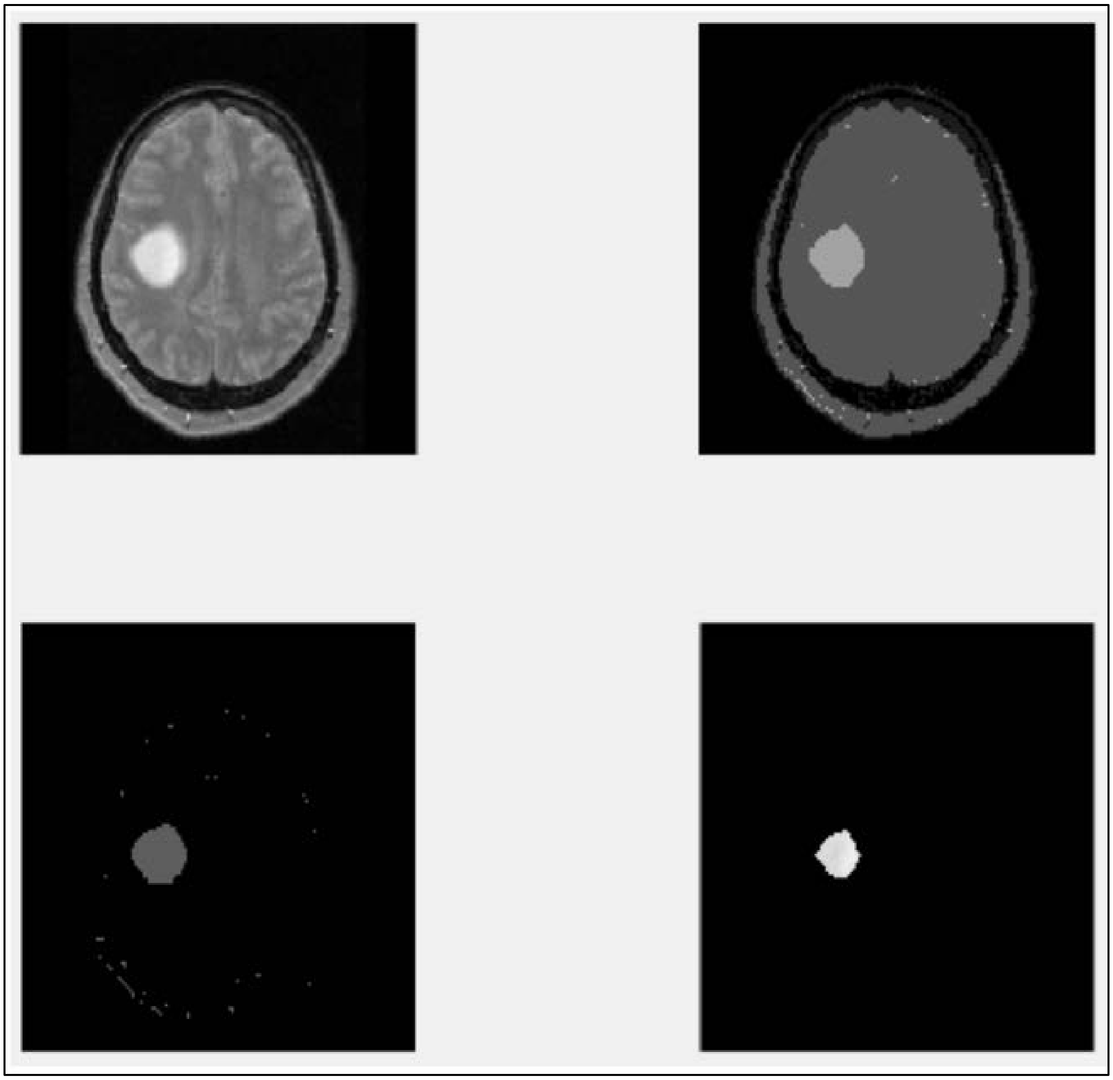


Figure 24: Output from PSO Algorithm (Benign)

4.1.3 Feature Extraction

Discrete Wavelet Transform (DWT) and Principal Component Analysis (PCA) with Gray-Level Co-occurrence Matrix Algorithm (GLCM)

The discrete wavelet transform (DWT) is a powerful implementation of the WT using the dyadic scales and positions. The fundamentals of DWT are introduced as follows. Suppose x(t) is a square-integrable function, then the continuous WT of x(t) relative to a given wavelet ψ(t) is defined as

$$W_{\psi}(a,b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}(t)dt \tag{1}$$

where

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-a}{b}\right) \tag{2}$$

Here, the wavelet ψa,b(t) is calculated from the mother wavelet ψ(t) by translation and dilation: a is the dilation factor and b the translation parameter (both real positive numbers). There are several different kinds of wavelets which have gained popularity throughout the development of wavelet analysis. The most important wavelet is the Harr wavelet, which is the simplest one and often the preferred wavelet in a lot of applications.

GLCM is a widely used method for medical image analysis, classification. This method gives us information about relative position of two pixels with respect to each other. The GLCM is then created by counting the number of occurrences of pixel pairs at a certain distance. To compute the GLCM matrix for an image f (i, j), a distance vector d=(x, y) is defined. The (i,j)th element of the GLCM matrix P is defined as the probability that grey levels i and j occur at distance d and angle θ, then extracting texture features from GLCM matrix P. Four angles (0,45,90,135) and four distances (1,2,3,4) can be used to calculate the co-occurrence matrix.

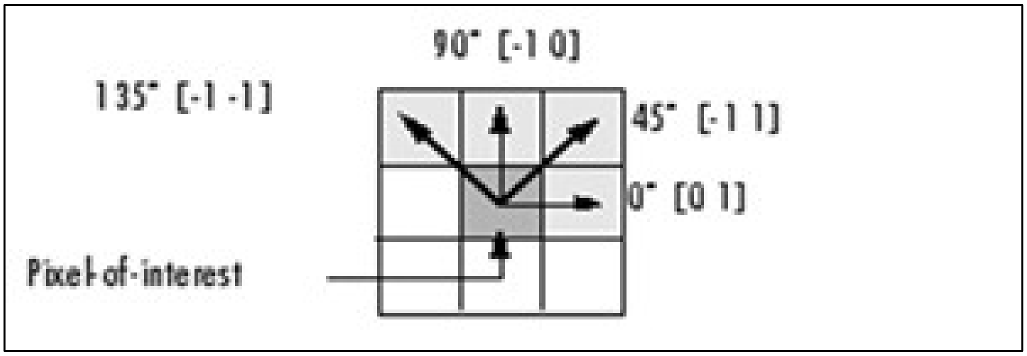


Figure 25: Calculation of co-occurrence matrix in GLCM

The extracted features are:

1. **Correlation:**

It measures the linear dependency of grey levels of neighboring pixels. It is defined in Eq.1

$$Correlation = \sum_{i,j=0}^{N-1} P_{ij} \frac{(i-\mu)(j-\mu)}{\sigma^2}$$

Equation 1: Correlation

2. **Contrast**

Also called the sum of Square Variance. It defers the calculation of the intensity contrast linking pixel and its neighbor over the whole image. It is defined in Eq. 2

$$Contrast = \sum_{i,j=0}^{N-1} P_{ij} (i-j)^2$$

Equation 2: Contrast

3. **Energy**

It makes use for the texture that calculates orders in an image. It gives the sum of square elements in GLCM. It is defined in Eq. 3

$$Energy = \sum_{i,j=0}^{N-1} \left(P_{ij}\right)^2$$

Equation 3: Energy

4. **Homogeneity**

It passes the value that calculates the tightness of distribution of the elements in the GLCM to the GLCM diagonal. It is defined in Eq. 4

$$Homogeneity = \sum_{i,j=0}^{N-1} \frac{P_{ij}}{1+(i-j)^2}$$

Equation 4: Homogeneity

5. **Mean**

Defined as the mean of the pixel values of the input image. It is defined in Eq. 5

$$Mean = \sum_{i,j=0}^{N-1} i(P_{i,j})$$

Equation 5: Mean

6. **Standard Deviation**

It is defined as the dispersion of the pixel in consideration from the mean of the pixels of the input image. It is defined in Eq. 6

$$Standard\ Deviation = \sqrt{\sigma_i^2}$$

Equation 6: Standard Deviation

7. **Entropy**

It shows the amount of information of the image that is needed for the image compression. Entropy measures the loss of information or message in a transmitted signal and also measures the image information. It is defined in Eq. 7

$$Entropy = \sum_{i,j=0}^{N-1} -\ln(P_{ij})P_{ij}$$

Equation 7: Entropy

8. **Root Mean Square (RMS)**

RMS is calculated on a set of pixels by taking the square of each pixel, calculating the sum of those squares, and taking the square root. The result is scaled by the number of pixels. RMS gives an accurate measurement of the amount of noise present. It is defined in Eq. 8

$$\text{RMS noise} = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \frac{\sum_{i=1}^{n} X_i}{n})^2}{n}}$$

Equation 8: Root Mean Square

9. **Variance**

It is the expectation of the squared deviation of a pixel from its mean. It is defined in Eq. 9

$$Variance = \sum_{i,j=0}^{N-1} P_{i,j}\,(i - \mu_i)^2$$

Equation 9: Variance

10. **Inverse Difference Movement (IDM)**

Inverse Difference Moment (IDM) is the local homogeneity. It is high when local gray level is uniform and inverse GLCM is high. It is defined in Eq. 10

$$IDM = \frac{\sum_{i=0}^{Ng-1}\sum_{j=0}^{Ng-1} P_{ij}}{1+(i-j)^2}$$

Equation 10: Inverse Difference Movement

4.1.4 Classification

4.1.4.1 Support Vector Machine (SVM)

Support vector machines (SVMs) are a type of supervised learning models along with associated learning algorithms that analyze data and recognize various patterns, used for classification analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes, malignant and benign forms the output, making it a non-probabilistic binary linear classifier. Now that there are set of training examples at hand, each marked as belonging to one of two categories, an SVM training algorithm constructs a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. Newer examples are then plotted into it and then predicted to belong to a category based on which side of the gap they fall on.

More formally, a support vector machine constructs a hyper plane or set of hyper planes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyper plane that has the largest distance to the nearest training data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.

A SVM takes a set of feature vectors as input, generates a training model after scaling, selecting and validating, and generates a training model as the output. The following figure represents the training process of a SVM:

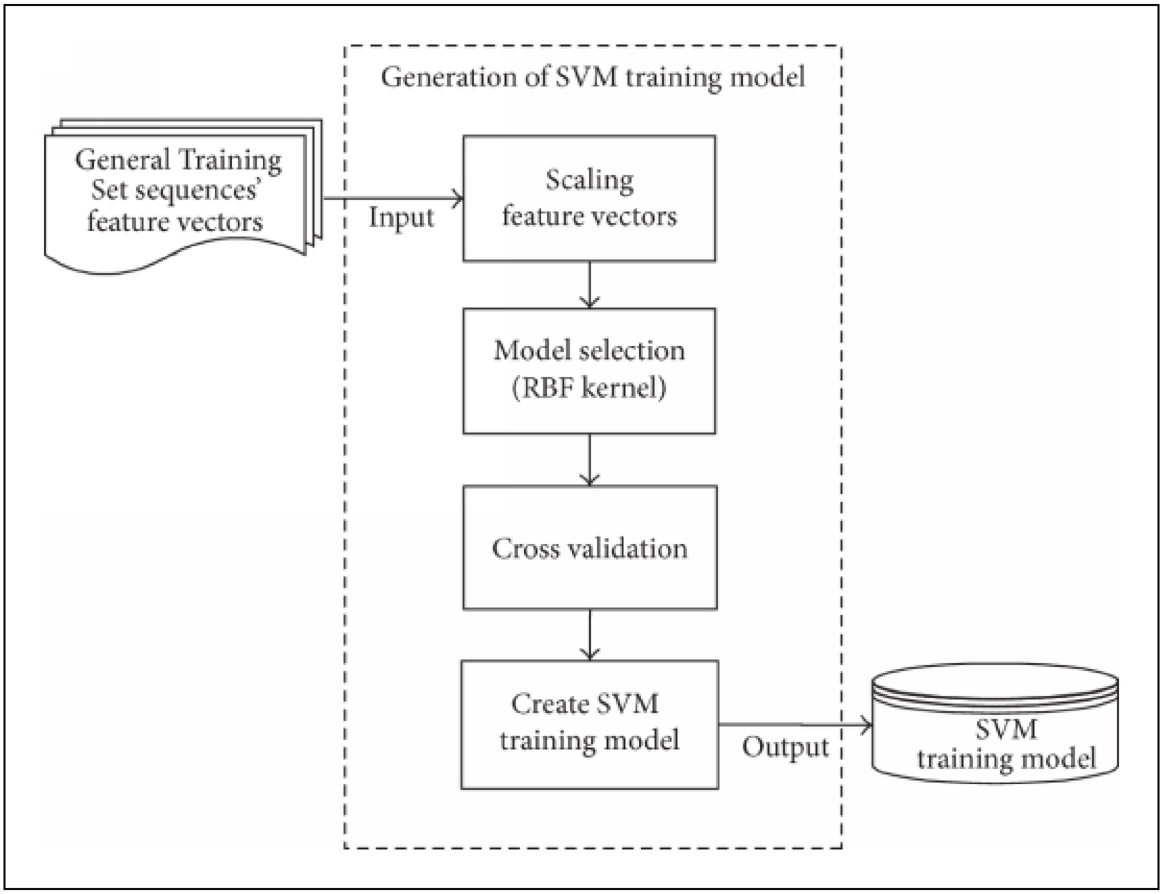


Figure 26: Training a SVM Model

4.1.4.2 Lazy IBK

Instance based learning algorithms compare the new problem instances with the instances that are seen during the training of the algorithm and are stored in memory. It is different from normal algorithms, as it does not perform explicit generalizations.

It is called instance-based because it constructs hypotheses directly from the training instances themselves. As the hypothesis grows so does the hypothesis complexity. In the worst case, a hypothesis is a list of n training items and the computational complexity of classifying a single new instance is O(n).

The methods for data-driven modelling considered up to now were first building a model of the available (training) data (process of learning), and then were put to operation, when classification or numerical prediction was taking place. These methods are sometimes referred to as eager learning (since they are eager to build a model first).

Instance-based (IB) learning methods simply store the training examples and postpone the generalization (building a model) until a new instance must be classified or prediction made. The model that is built by IB methods is not a global model that uses all training data, but rather a local model involving only some of the instances. The IB methods are used both for classification and for regression. Most important methods are: nearest neighbor method, locally weighted regression, and case-base reasoning. Other names for IB methods are: exemplar-based, case-based, experience-based, edited k-nearest neighbor.

The instance based k-nearest neighbor is a semi-supervised learning algorithm. It requires training data and a predefined k value to find the k nearest data based on distance computation for each instance. If k data have different classes, the algorithm predicts class of the unknown data to be the same as the majority class.

Given an mx-by-n data matrix X, which is treated as mx (1-by-n) row vectors x1, x2, ..., xmx, and my-by-n data matrix Y, which is treated as my(1-by-n) row vectors y1, y2,..., ymy, the various distances between the vectors xs and yt are Euclidean distance, Hamming distance and Cosine distance.

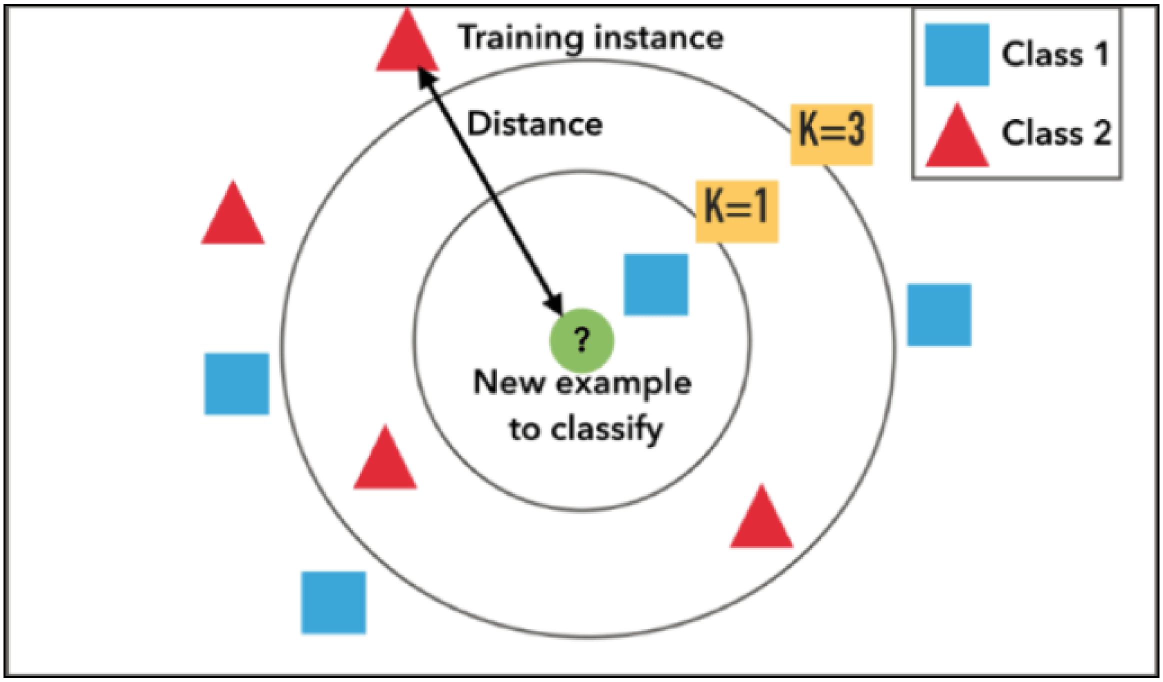


Figure 27: Instance Based KNN

4.1.4.3 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are similar to traditional neural networks which are made of neurons with learnable weights and biases. Each neuron receives several inputs, takes a weighted sum over them, pass it through an activation function and responds with an output [21].

The CNN algorithm is a multilayer perceptron that is the special design for identification of two-dimensional image information. It has the following layers: input layer, convolution layer, sample layer and output layer. Each neuron parameter is set to the same parameter, namely, the sharing of weights, i.e. each neuron with the same convolution kernels to deconvolution image. The figure below illustrates the process applied during CNN [21].

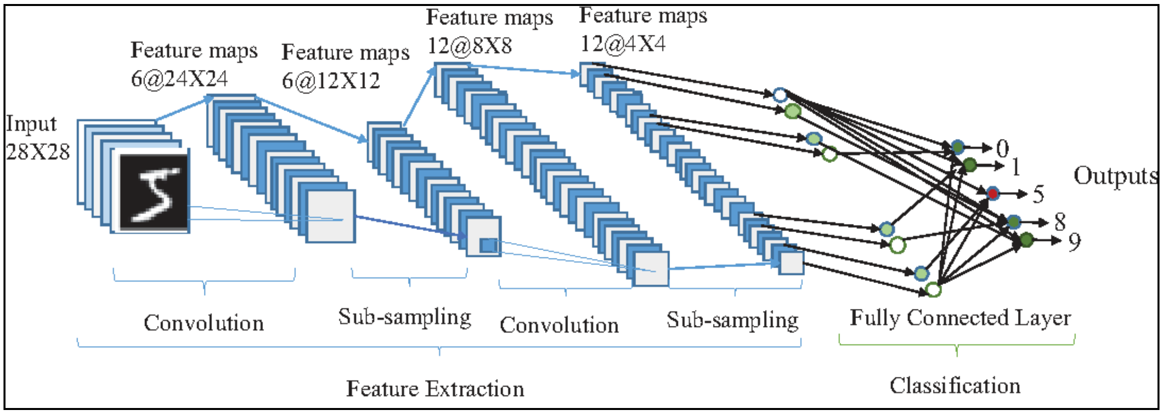


Figure 28: CNN Block Diagram

The CNN algorithm has two main processes: convolution and sampling.

The convolution process use a trainable filter F_x for the deconvolution of the input image (the first stage is the input image, the second input after the convolution is the feature image of each layer, namely the feature map), and then adds a bias b_x, which results in the convolution layer C_x.

The sampling process puts n pixels of each neighborhood through pooling steps to become a pixel, and then weighting by scalar weighting $W_x + 1$, adding the bias $b_x + 1$, and then the activation function produces a narrow n time feature map $S_x + 1$.

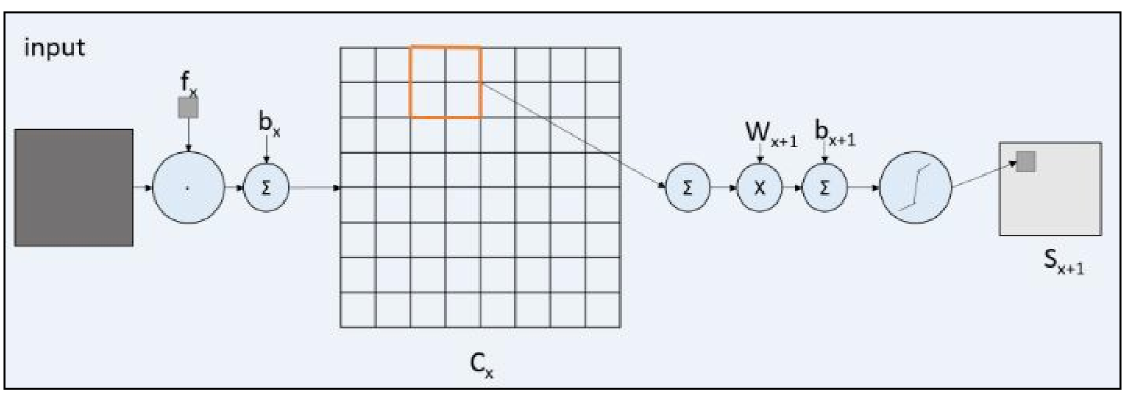


Figure 29: Working of CNN

The main idea of the CNN algorithm is the usage of the local receptive field, the weight sharing, and the sub sampling by time or space for feature extraction and reducing the size of the training parameters. CNN is advantageous as it explicitly avoids feature extraction and tries to implicitly learn from the training data.

The neuron weights are the same on the surface of the feature mapping, thus enabling the network to learn in parallel, and helping to reduce the complexity of the network. Further, adopting the sub sampling structure by time or space, can achieve some degree of robustness, scale and deformation displacement.

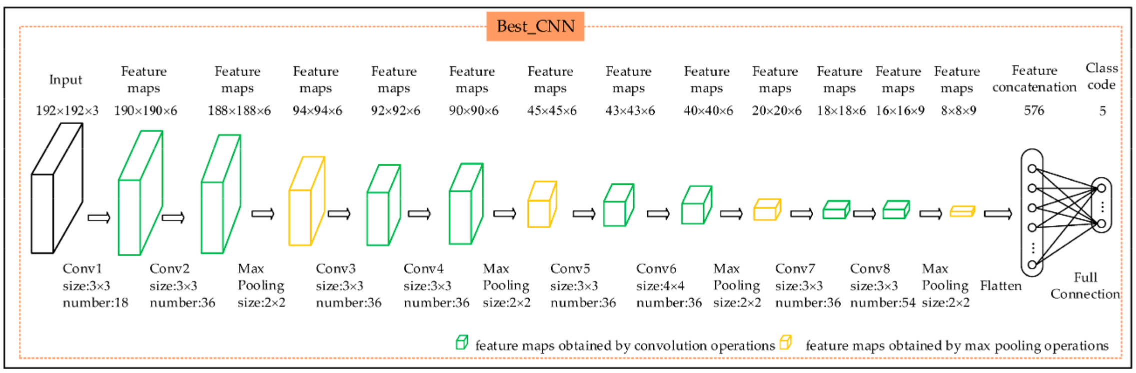


Figure 30: The CNN Process

4.2 Programming Language Used for Implementation

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages.

4.3 Tools Used

- Neural Network ToolBox.
- Languages: MATLAB.
- Documentation: Microsoft Word, Google docx.
- Software: MATLAB 2017b
- Hardware: Intel Dual Core Dual Processor or advanced version; Minimum 256 MB of RAM; Minimum 1 GB of Hard disk Space

4.4 Testing Approach

The testing of the system is done through a process of different tests to ensure the correct working of the software and measure its capabilities and limitations. A brief explanation of the proposed type of tests to be conducted are given below:

<u>Unit Testing</u>

- It concentrates on the efforts required for verification on the minute units of software design which namely, is the software module.
- We use the component level design description as a guide. Important control paths are tested to uncover errors within the boundary of the module.
- The unit test is white box oriented, and the steps can be conducted in parallel for multiple modules.

<u>Integration Testing</u>

- Interfacing of various modules can be problematic.
- Data loss can occur across an interface, one module may affect the other, and individually acceptable imprecision may be magnified when combined.
- Integration testing is thus used to construct the program structure and also to conduct tests to unfold errors related to the interface.

<u>Stress Testing</u>

- Stress tests are conducted to confront programs with abnormal situations.
- Stress testing forces a system to execute in a manner that demands for resources in abnormal quantity, frequency or volume which allows us to gauge the limit of the system.

Performance Testing

- Performance testing is conducted through all the steps in the testing process to test runtime performance of software within the context of an integrated system.

Security Testing

- This system manages sensitive information related to patients. There may be causes and actions that can harm these individuals thus becoming a target for improper or illegal penetration.
- Security testing attempts to verify that protection mechanisms built into a system will, in fact, protect it from improper penetration.
- During security testing, the tester plays the role of the hacker who desires to penetrate the system.
- Given enough time and resources, good security testing will ultimately penetrate the system. The role of the system designer is make penetrate cost more than value of the information that will be obtained.

Recovery Testing

- Many computers – based systems must recover from faults and resume processing within a pre – specified time. In some cases, a system must be fault tolerant, i.e. processing faults must not cause overall system function to cease. In other cases, a system failure must be corrected within a specified period of time or severe economic damage will occur.
- The testing approach that was followed in the project began at the preliminary level and this testing was worked outwards toward the integration of the entire system.
- Testing approach was an umbrella activity in the development of the system. Each module was tested at its completion before transitioning to the next component.

- After selecting a particular data set for a kidney tumor and selecting some probably suitable combinations for the network, they were tested and the output was compared with actual desired output.
- After the development of GUI the integration of network was tested with the GUI.
- Different types of testing adopted in the approach are as follows: Unit testing, Integration testing

4.5 Testing Plan

Unit Testing

Each module will be tested to check whether it gives the desired output. The segmentation algorithms will be executed and the output images will be checked for the segmented tumor. In case of feature extraction, it will be ensured that the features are stored in an excel sheet and don’t result in NaN values.

The classification algorithms will be trained on the datasets and then tested on the training datasets to obtain the accuracy of the classifier.

Integration Testing

All the possible combinations of the individual modules will be integrated and tested to ensure that it produces correct output. Each segmentation module will be combined with feature extraction module and then combined with classification module for testing.

System Testing

The GUI will be tested in system testing. The input will be given to the system through the GUI and the outputs will be checked.

Test Schedule

Test Title	Date
Test the selected data set with segmentation algorithm.	1st October 2017
Test the selected data set with feature extraction algorithm.	30th October 2017
Test the selected data set with classification algorithm.	15th January 2018
Test the selected data set with all combinations of segmentation, feature extraction and classification algorithms.	14th February 2018
Test the whole system with the GUI	2nd April 2018

Table 4: Testing Schedule

Unit Test Cases

Test Case:	Average Accuracy
SVM Accuracy	84%
Lazy IBK Accuracy	82%
CNN Accuracy	88%

Table 5: Unit Test Cases

Integration Test Cases

Test Case:	**Accuracy**
OTSU + (DWT + PCA) + SVM	56%
OTSU + (DWT + PCA) + Lazy IBK	45%
OTSU + CNN	87%
CANNY + (DWT + PCA) + SVM	71%
CANNY + (DWT + PCA) + Lazy IBK	45%
CANNY + CNN	83%
PSO + (DWT + PCA) + SVM	45%
PSO + (DWT + PCA) + Lazy IBK	45%
PSO + CNN	91%
Watershed + (DWT + PCA) + SVM	63%
Watershed + (DWT + PCA) + Lazy IBK	44%
Watershed + CNN	76%

Table 6: Integration Test Cases

4.6 Results

Confusion Matrix

The output of the computing done via all the classification algorithms is mapped via the confusion matrix. A confusion matrix consist of information about actual and predicted classes done by a classification system. Performance of such systems is evaluated using the data in the matrix. The following table shows the confusion matrix for a two class classifier. Confusion Matrix helps in detecting the accuracy of datasets.

The data in the confusion matrix shown has the following meaning in context.

		Predicted	
		Negative	Positive
Actual	Negative	**a**	**b**
	Positive	**c**	**d**

Table 7: Confusion Matrix Calculation

- a is the number of correct predictions that an instance is negative,
- b is the number of incorrect predictions that an instance is positive,
- c is the number of incorrect of predictions that an instance negative, and
- d is the number of correct predictions that an instance is positive.

The results for one run of the above testing are shown here in the form of confusion matrices for each possibility illustrated in Table 6.

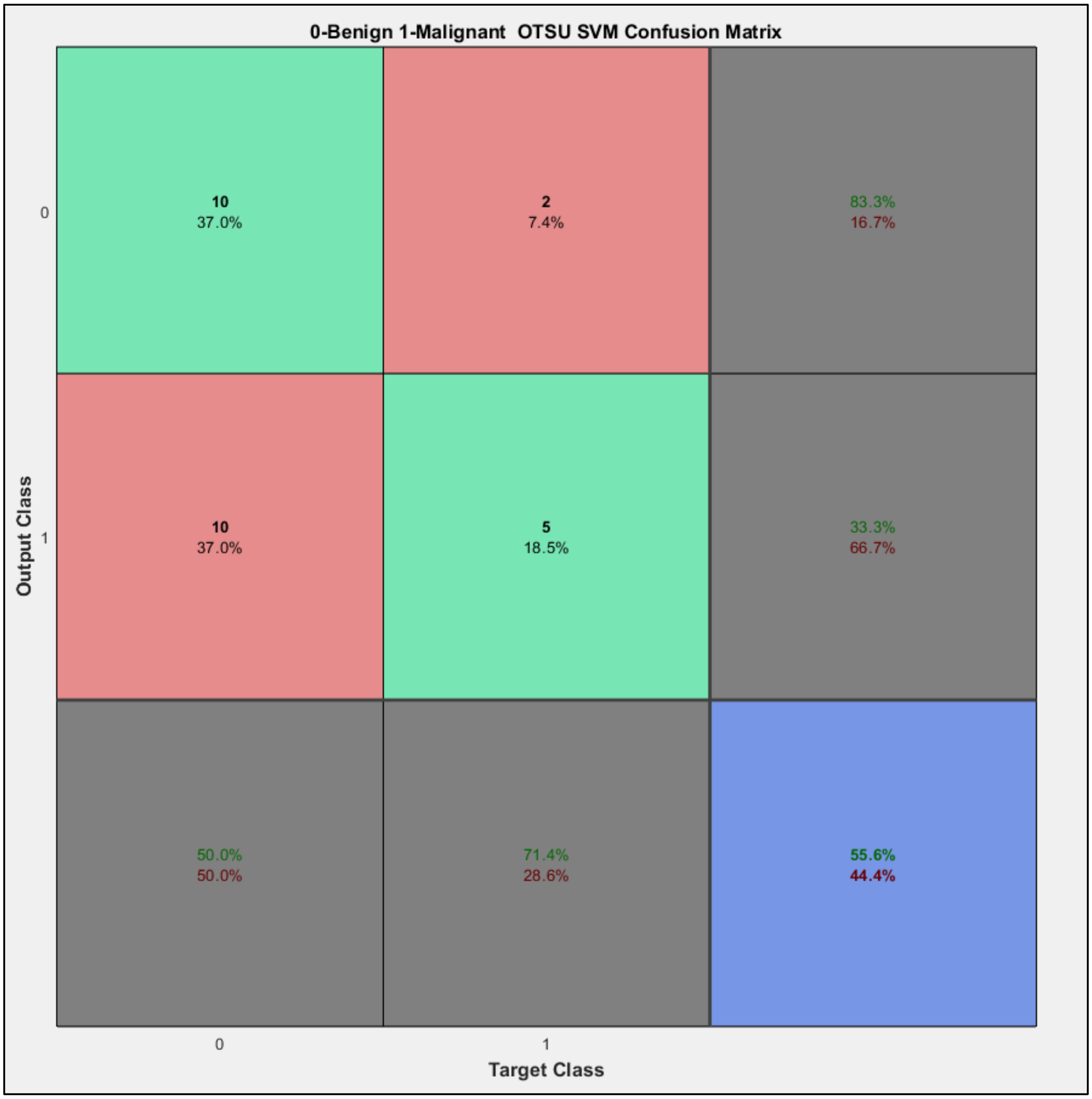


Figure 31: Confusion Matrix for OTSU and SVM

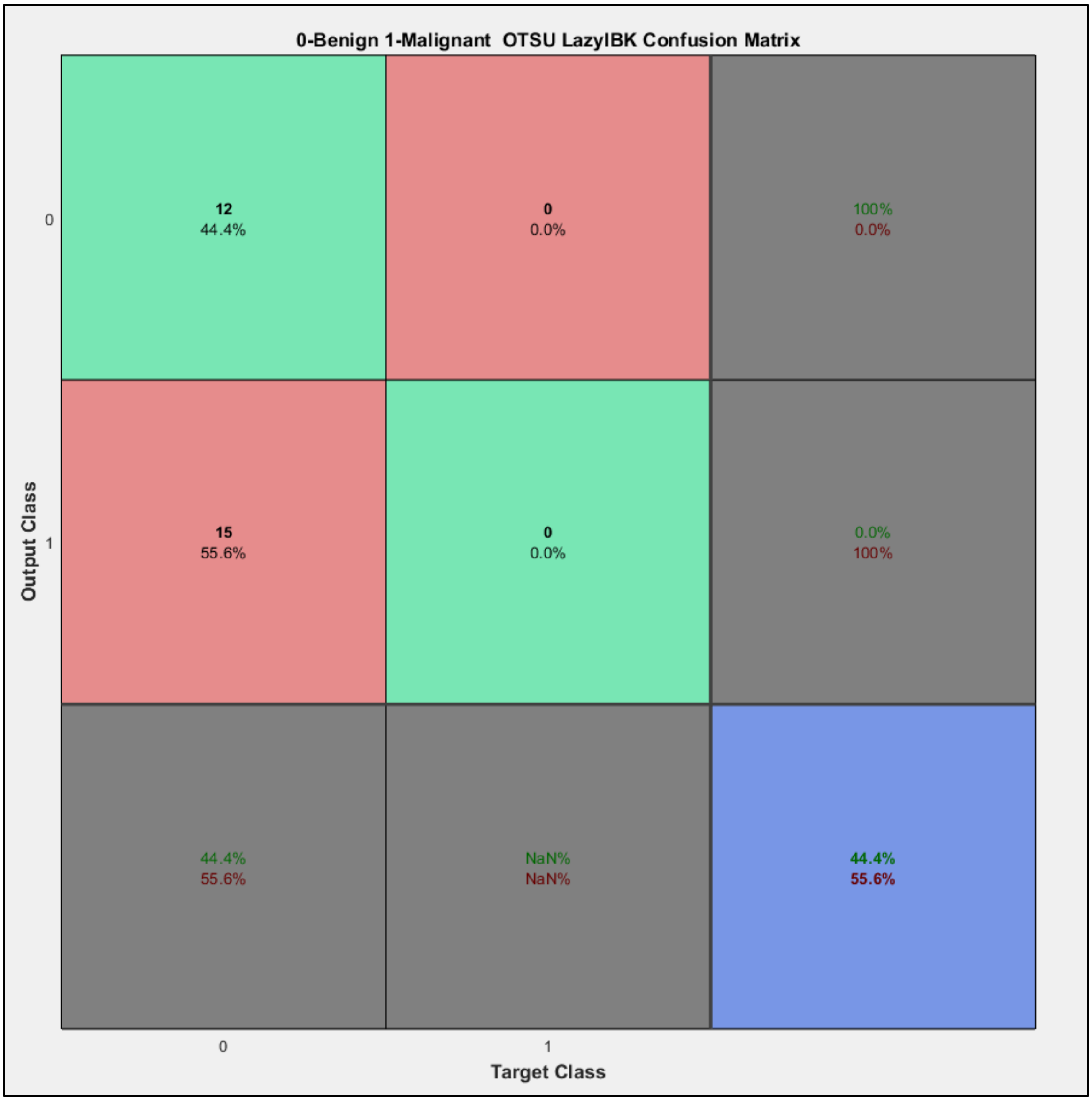


Figure 32: Confusion Matrix for OTSU and Lazy IBK

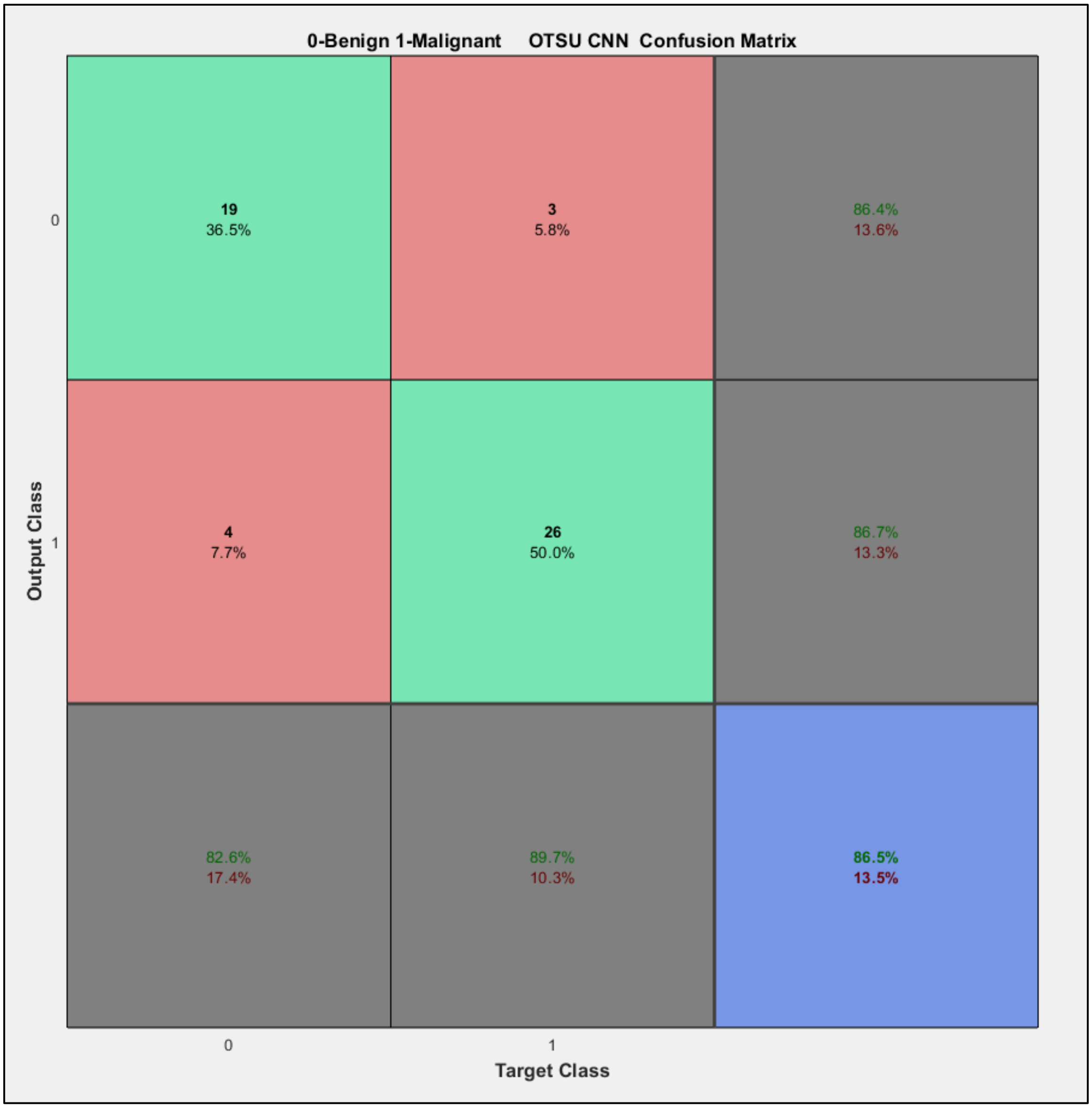


Figure 33: Confusion Matrix for OTSU and CNN

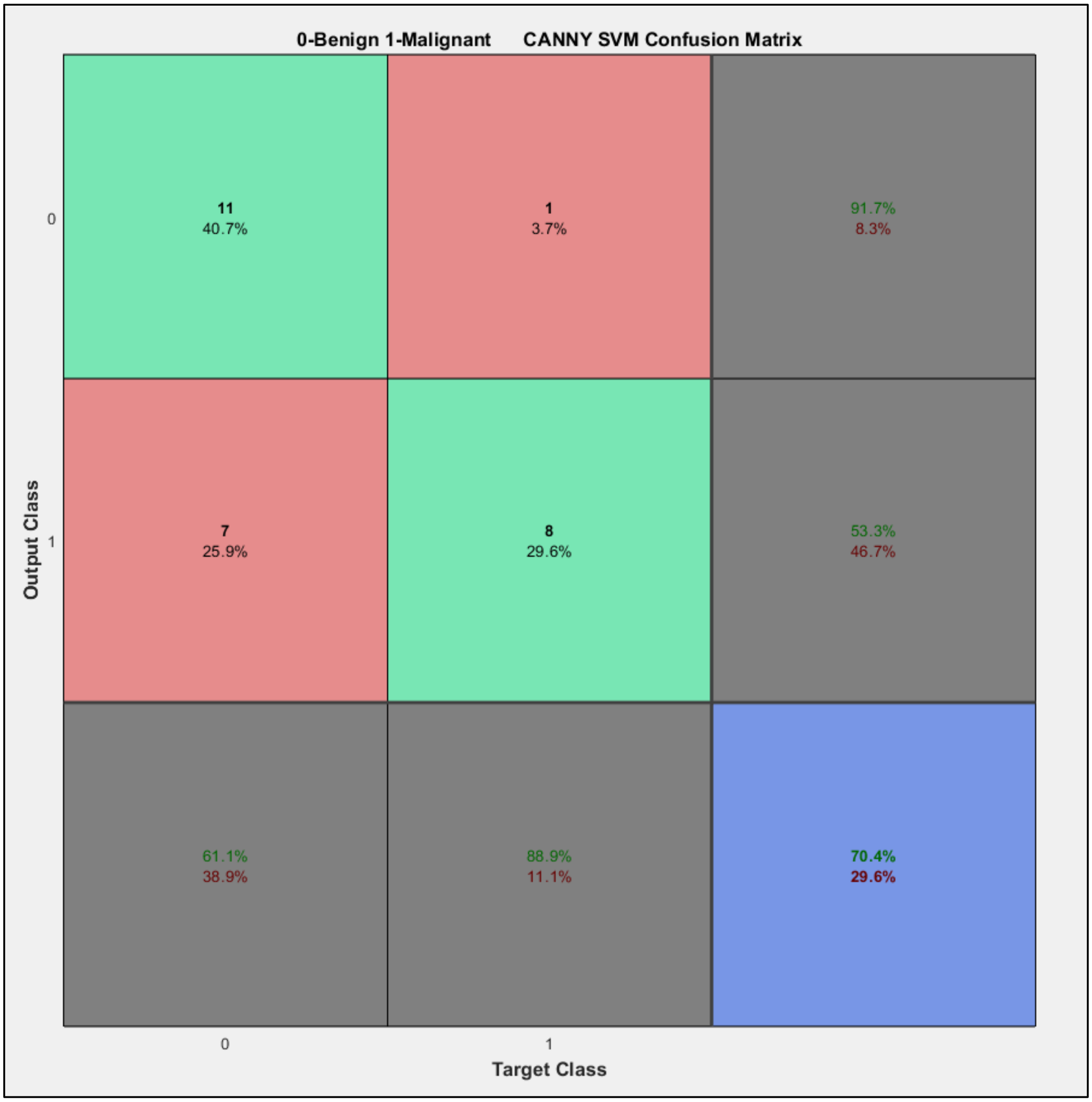


Figure 34: Confusion Matrix for Canny and SVM

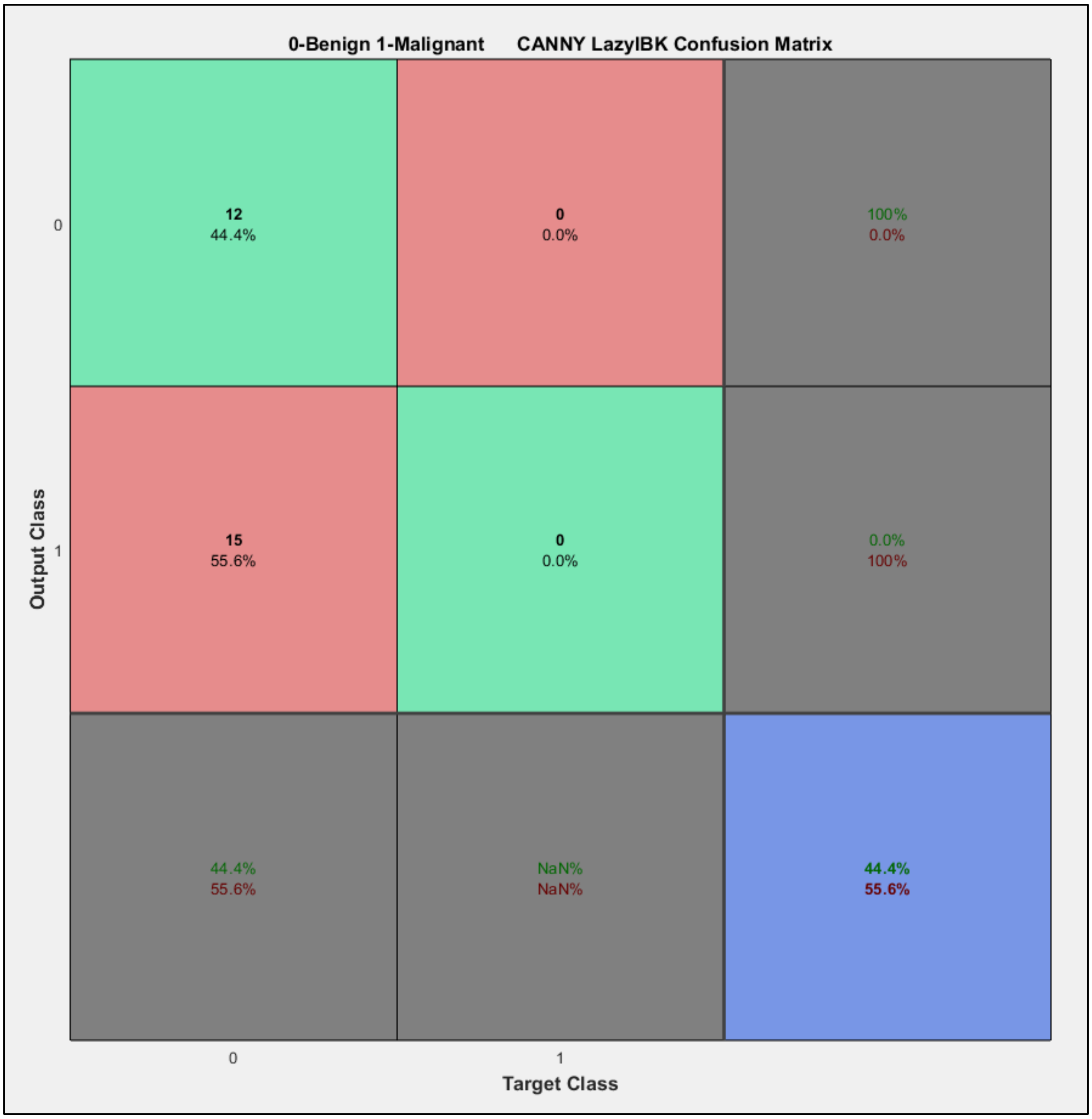


Figure 35: Confusion Matrix for Canny and Lazy IBK

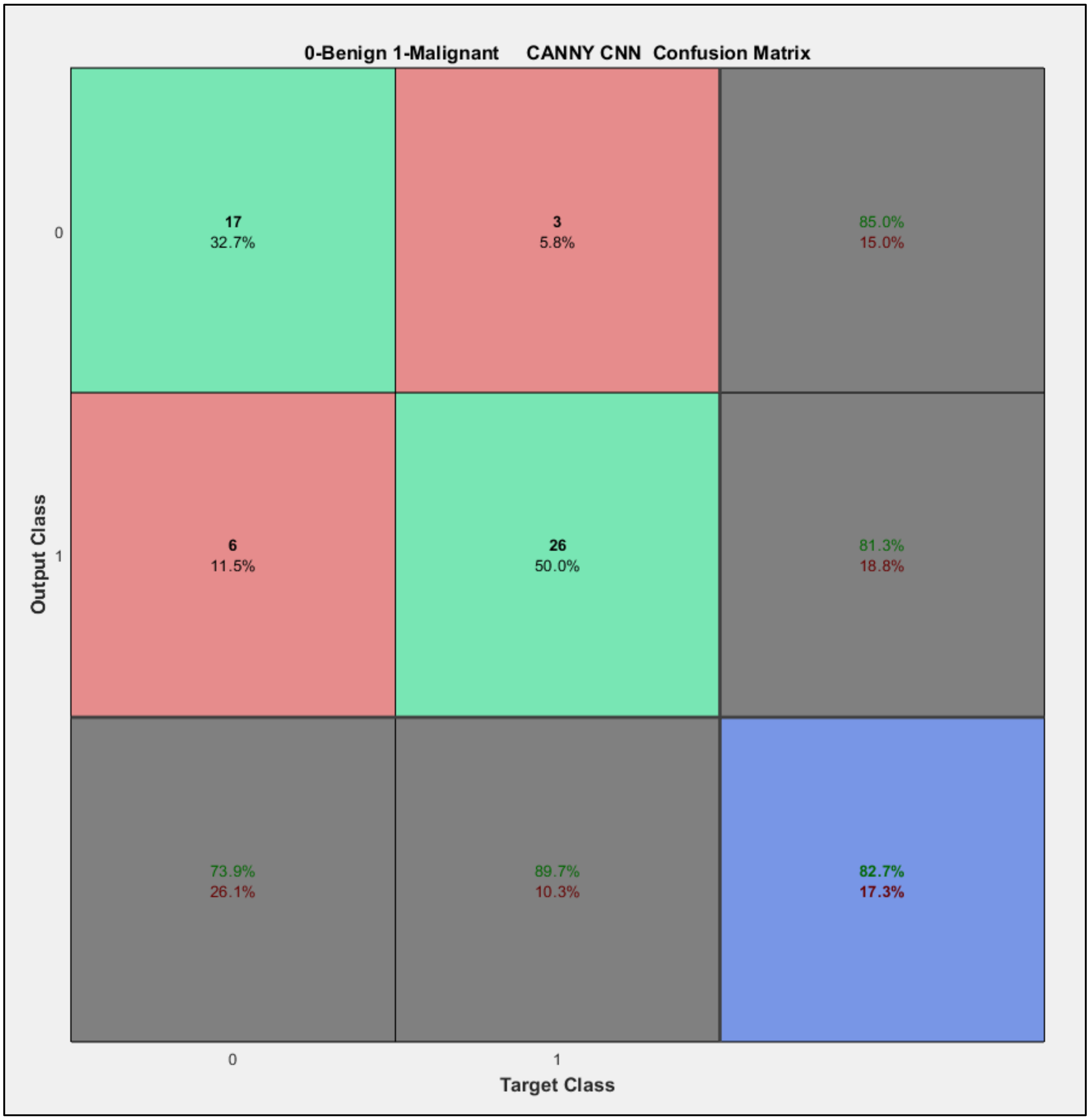


Figure 36: Confusion Matrix for Canny and CNN

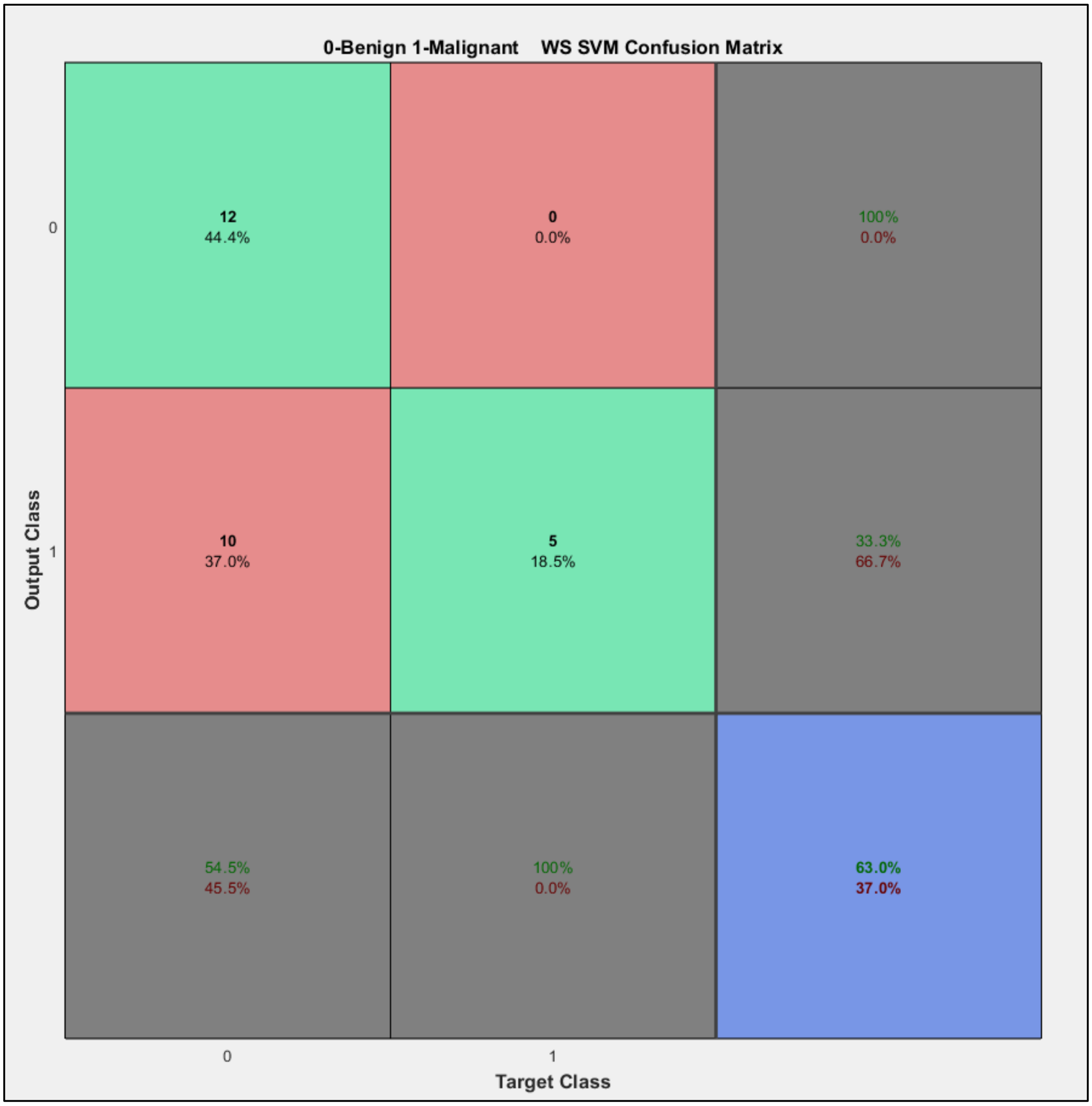


Figure 37: Confusion Matrix for Watershed and SVM

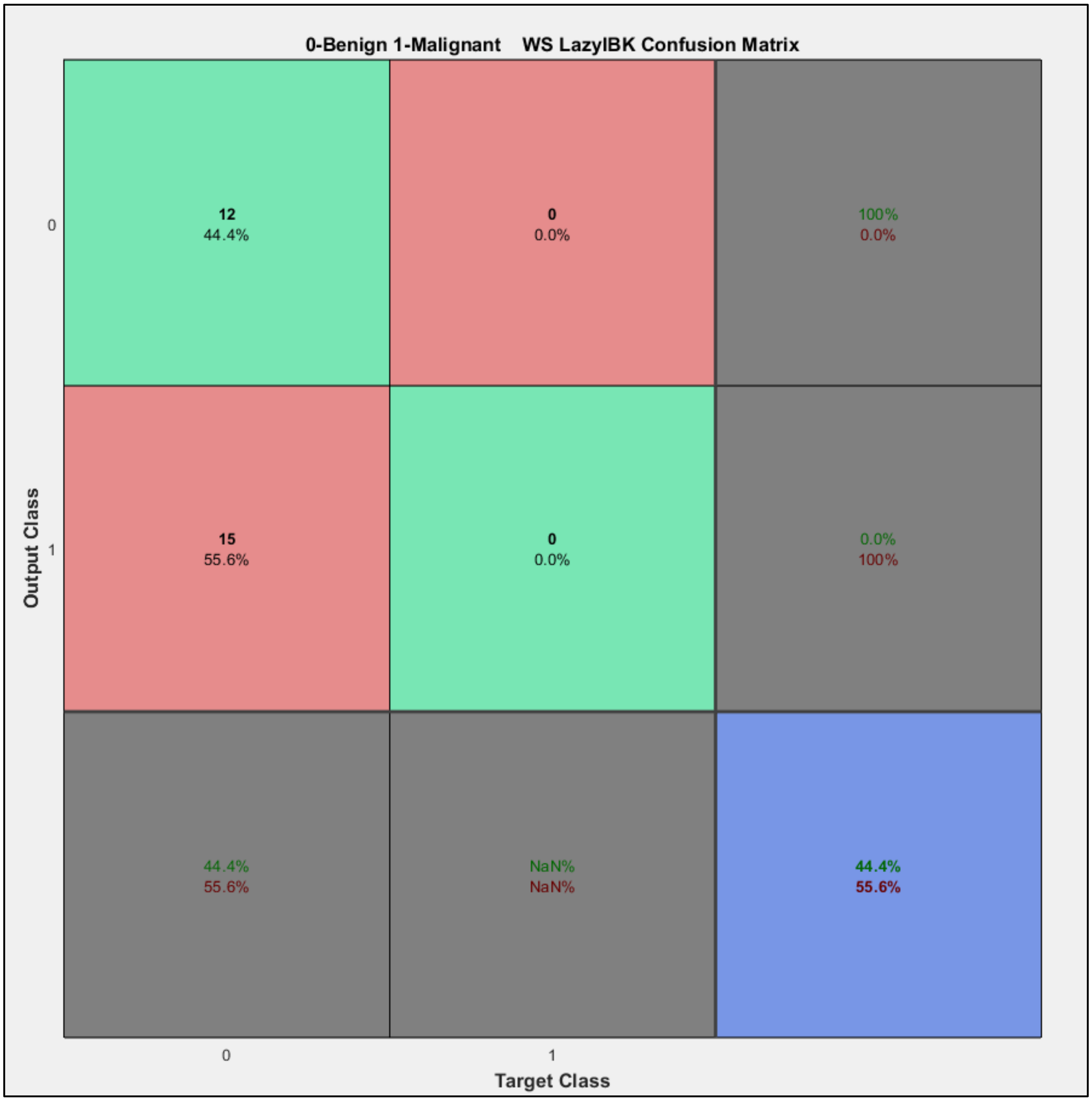


Figure 38: Confusion Matrix for Watershed and Lazy IBK

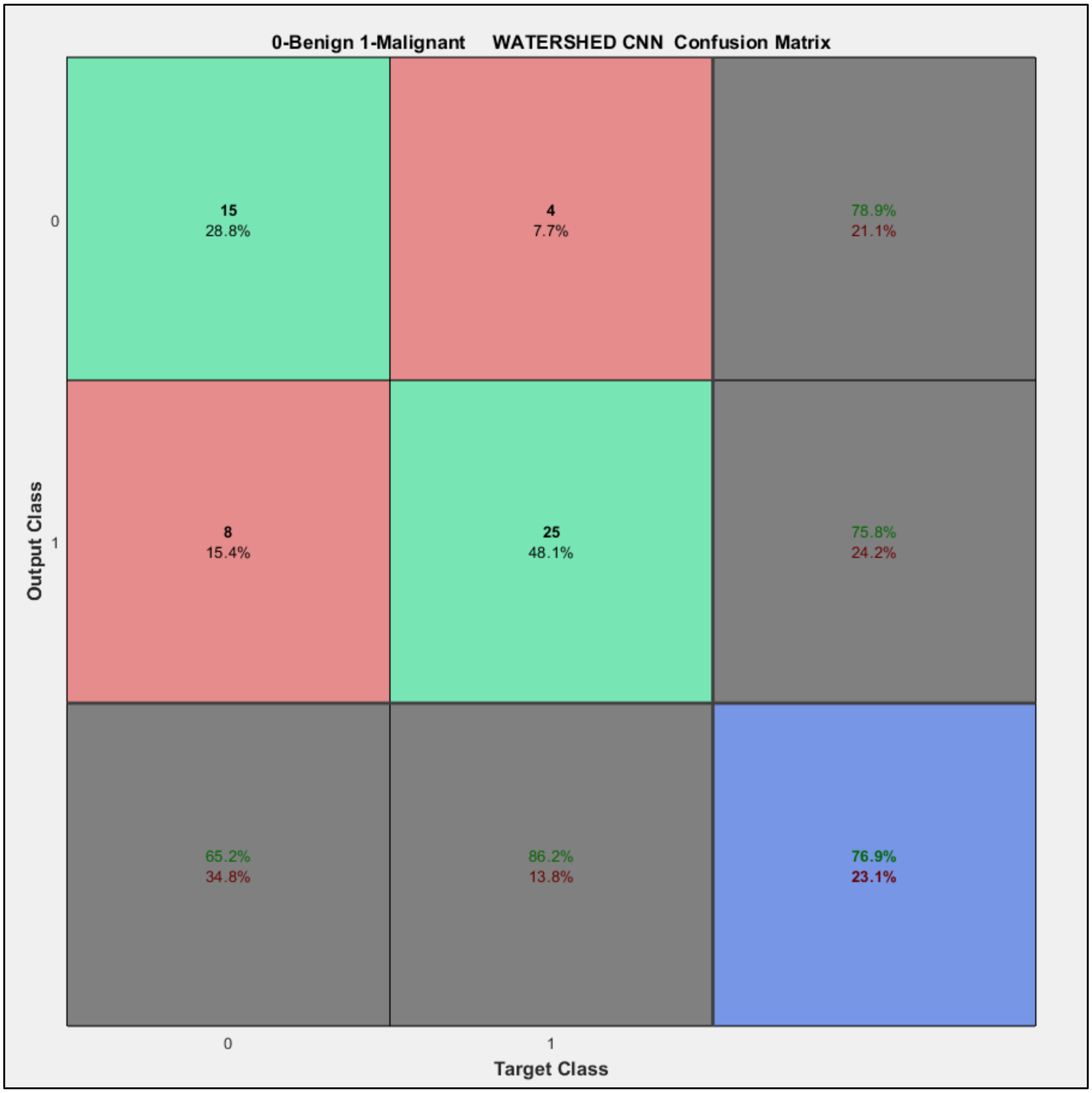


Figure 39: Confusion Matrix for Watershed and CNN

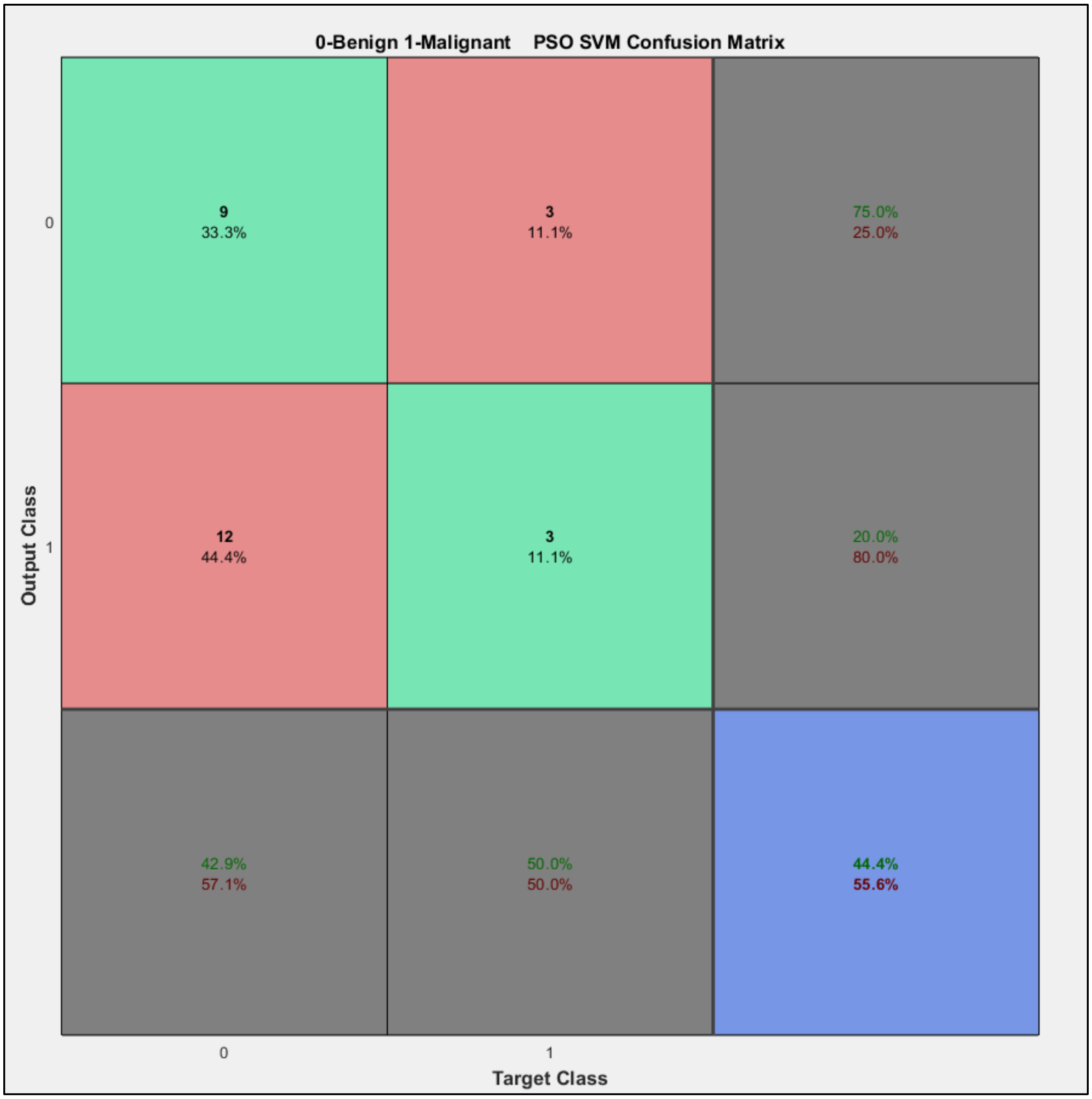


Figure 40: Confusion Matrix for PSO and SVM

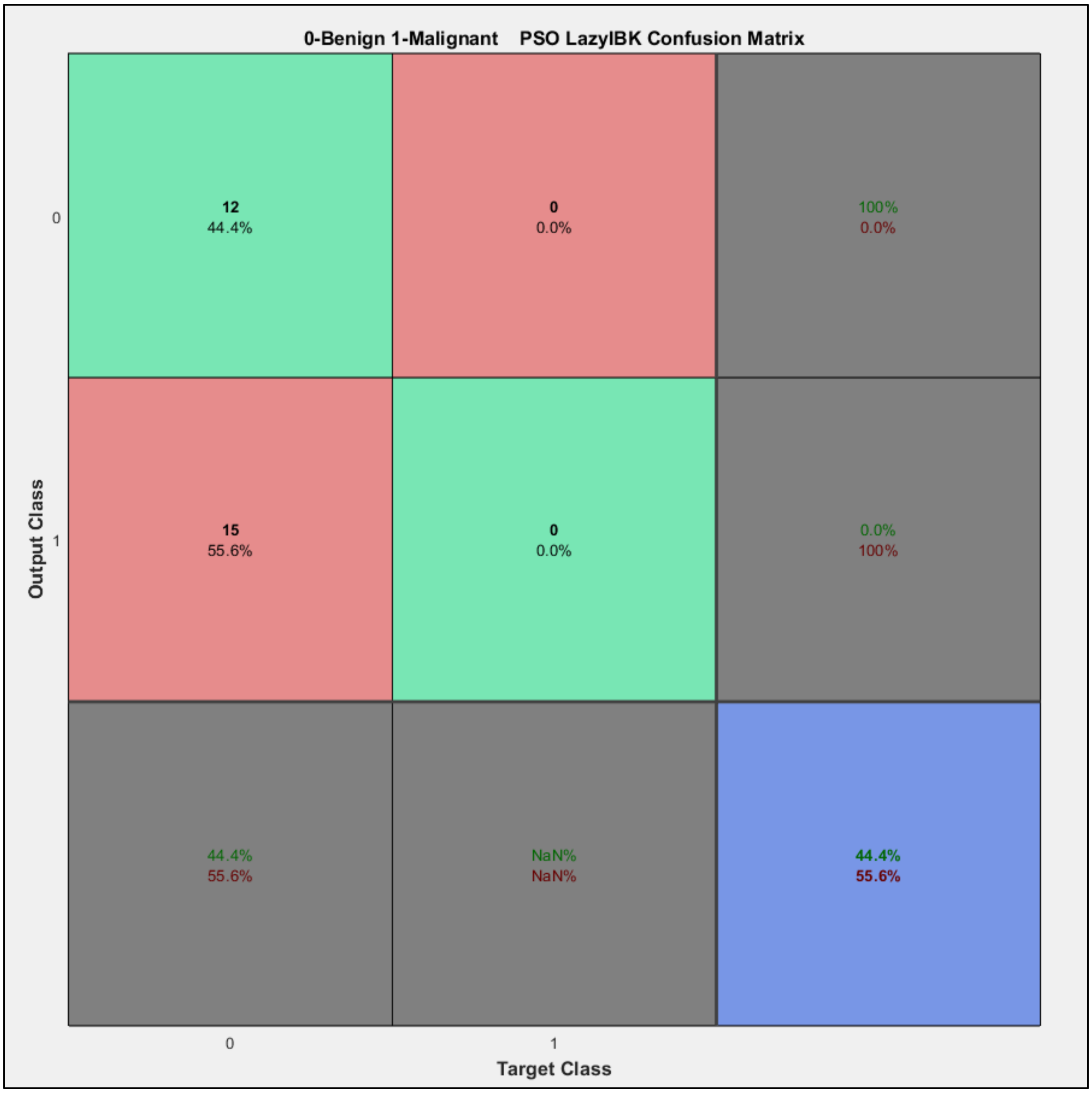


Figure 41: Confusion Matrix for PSO and Lazy IBK

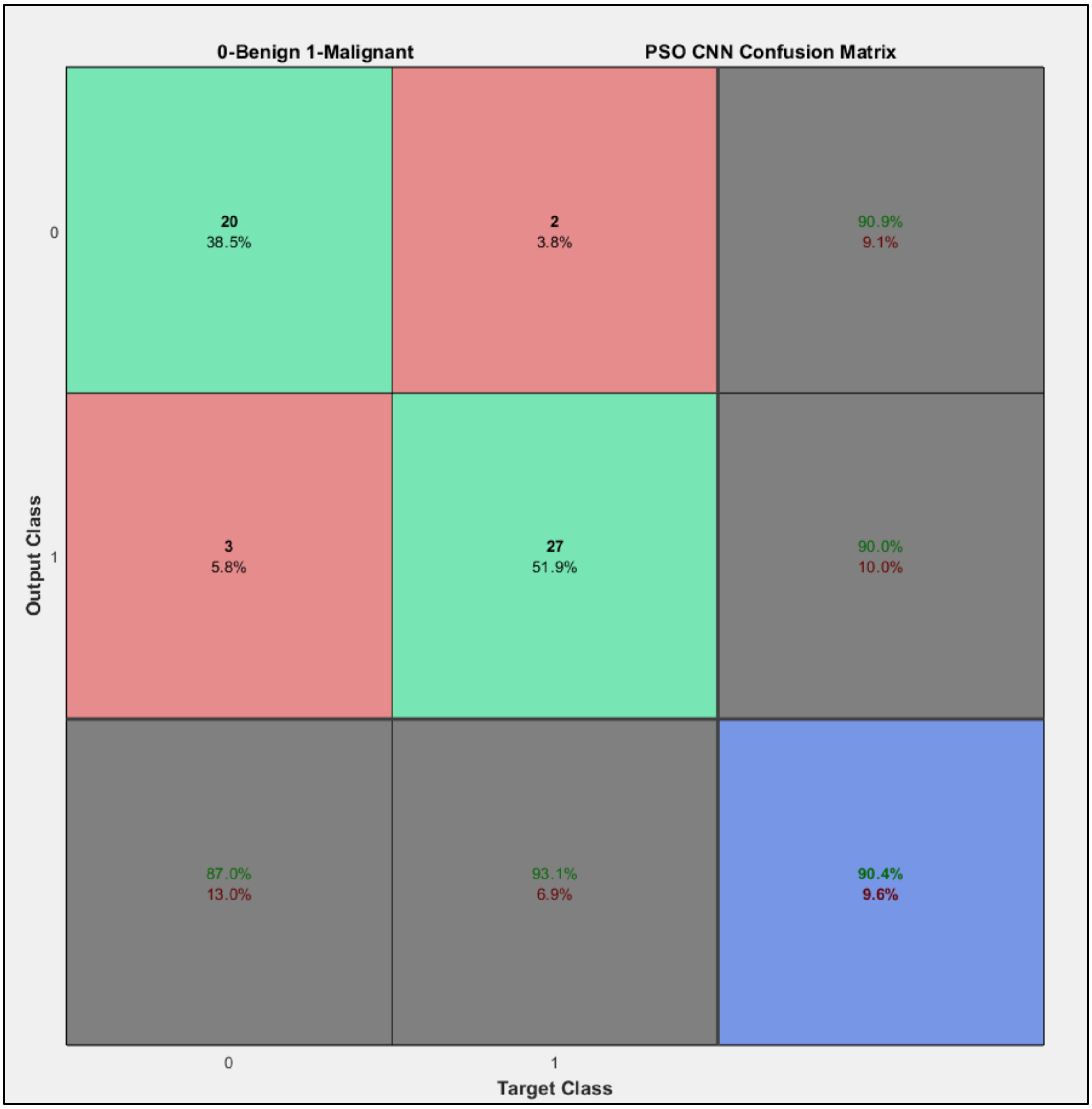


Figure 42: Confusion Matrix for PSO and CNN

Chapter 5. Conclusion and Future Scope

This chapter discusses the lessons learned and the knowledge gained after the completion of our project and the possible future scope of our project.

5.1 Conclusion

Abnormal growth of tissue in the brain which affect the normal functioning of the brain is considered a brain tumor. The main goal of medical image processing is to identify accurate and meaningful information using algorithms with minimum error possible. Brain tumor detection and classification through MRI images can be categorized into four different sections: pre-processing, image segmentation, feature extraction and image classification. Various segmentation methodologies are explored in the project. It can be concluded that the algorithms and the parameters used in the proposed system are all meant to increase the efficiency of the system by achieving better results.

The boundary approach and the edge based approach for segmentation are very common but the region growing approach gives better results. It is found that the particle swarm optimization algorithm gives the most accurately segmented tumors. Features extracted by using GLCM method help to increase efficiency as minute details of tumor by using various features can be extracted. Of the various classification methods studied, it was experimentally found that the convolution neural networks give the best classification accuracy. Accuracy and reliability are of utmost importance in tumor diagnosis, as a patient's life depends on the results predicted by the system. Thus, the proposed methodology helps in increasing the accuracy and obtaining the desired results.

5.2 Future Scope

Encouraged by these results, future work will involve the improvement of classification result and overall accuracy. The number of output classes can also be increased if more data is available. With a more extensive and diverse dataset, the overall classification accuracy can be dramatically increased.

Another approach to improve the result would be to increase the number of hidden layers of the neural network. By increasing the number of hidden layers, the weights will be better adjusted and thus increase the classification. One can also do fine tuning and transfer learning approaches to better tune the model on the basis of already trained models.

References

[1] R. Mishra, “MRI based brain tumor detection using wavelet packet feature and artificial neural networks,” *Proceedings of the International Conference and Workshop on Emerging Trends in Technology - ICWET 10*, 2010.

[2] Y. K. Dubey and M. M. Mushrif, “Segmentation of brain MR images using intuitionistic fuzzy clustering algorithm,” *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing - ICVGIP 12*, 2012.

[3] S. M. K. Hasan, M. Ahmad, and S. D. Ghosh, “Perceptive Proposition of Combined Boosted Algorithm for Brain Tumor Segmentation,” *Proceedings of the International Conference on Advances in Information Communication Technology & Computing - AICTC 16*, 2016.

[4] J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*. (2012)

[5] E. Küçükkülahli, P. Erdoğmuş, and K. Polat, “Brain MRI Segmentation based on Different Clustering Algorithms,” *International Journal of Computer Applications*, vol. 155, no. 3, pp. 37–40, 2016.Mohsen, Fahd, et al. "A new image segmentation method based on particle swarm optimization." *Int. Arab J. Inf. Technol.* 9.5 (2012): 487-493.

[6] Mohsen, Fahd, et al. "A new image segmentation method based on particle swarm optimization." *Int. Arab J. Inf. Technol.* 9.5 (2012): 487- 493.

[7] R. V. Rao, “Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,” *International Journal of Industrial Engineering Computations*, pp. 18–34, 2016.

[8] M. Al-Ayyoub, G. Husari, O. Darwish, and A. Alabed-Alaziz, “Machine learning approach for brain tumor detection,” *Proceedings of the 3rd International Conference on Information and Communication Systems - ICICS 12*, 2012.

[9] Q. Sun and H. Tian, “Interactive image segmentation using power watershed and active contour model,” *2012 3rd IEEE International Conference on Network Infrastructure and Digital Content*, 2012.

[10] I. Maiti and M. Chakraborty, “A new method for brain tumor segmentation based on watershed and edge detection algorithms in HSV colour model,” *2012 National Conference On Computing And Communication Systems*, 2012.

[11] Y. Liu and Q. Zhao, “An improved watershed algorithm based on multi- scale gradient and distance transformation,” *2010 3rd International Congress on Image and Signal Processing*, 2010.

[12] Kaur, Mandhir, and Rinkesh Mittal. "Survey of Intelligent Methods for Brain Tumor Detection." *International Journal of Computer Science Issues (IJCSI)* 11.5 (2014): 108.

[13] K. Hu, G.-Y. Tang, D.-P. Xiong, and Q. Qiu, “A novel image segmentation algorithm based on Hidden Markov Random Field model and Finite Mixture Model parameter estimation,” *2012 International Conference on Wavelet Analysis and Pattern Recognition*, 2012.

[14] D. Sridhar and I. V. M. Krishna, “Brain Tumor Classification using Discrete Cosine Transform and Probabilistic Neural Network,” *2013 International Conference on Signal Processing , Image Processing & Pattern Recognition*, 2013.

[15] A. Ladgham, G. Torkhani, A. Sakly, and A. Mtibaa, “Modified support vector machines for MR brain images recognition,” *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, 2013.

[16] P. Su, Z. Xue, L. Chi, J. Yang, and S. T. Wong, “Support vector machine (SVM) active learning for automated Glioblastoma segmentation,” *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012.

[17] H. Li and Y. Fan, “Label propagation with robust initialization for brain tumor segmentation,” *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012.

[18] J. Liu and L. Guo, “A New Brain MRI Image Segmentation Strategy Based on K-means Clustering and SVM,” *2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2015.

[19] R. Lang, L. Zhao, and K. Jia, “Brain tumor image segmentation based on convolution neural network,” *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP- BMEI)*, 2016.

[20] Akilandeswari, U., R. Nithya, and B. Santhi. "Review on feature extraction methods in pattern classification." *European Journal of Scientific Research* 71.2 (2012): 265-272.

[21] Liu, Tianyi, et al. "Implementation of training convolutional neural networks." *arXiv preprint arXiv:1506.01195* (2015).

Author’s Publications

Papers Published

Apurva Mehta, Hitesh Jaiswal, Poonam Bhogle, Shreyansh Kotak and Yash Pasar, “Brain Tumor Detection and Classification – A Survey”, *2018 5th IEEE International Conference on Computing for Sustainable Global Development*, *2018.*

Paper Presented

“Brain Tumor Detection and Classification – A Survey”, for an International Conference - 12th INDIACom, International Conference on Computing for Sustainable Global Development, 2018 presented by IEEE Delhi Section, CSI Delhi Chapter, ISTE Delhi Section and IETE Delhi Center on 15th March 2018.

Copy of Published Paper:

Proceedings of the 12th INDIACom; INDIACom-2018; IEEE Conference ID: 42835
2018 5th International Conference on "Computing for Sustainable Global Development", 14th - 16th March, 2018
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

Brain Tumor Detection and Classification – A Survey

Apurva Mehta
Student,
Department of Computer Engineering,
K.J. Somaiya College of Engineering,
Mumbai, India.
apurva.sm@somaiya.edu

Hitesh Jaiswal
Student,
Department of Computer Engineering,
K.J. Somaiya College of Engineering,
Mumbai, India.
hitesh.jaiswal@somaiya.edu

Poonam Bhogle
Assistant Professor,
Department of Computer Engineering,
K.J. Somaiya College of Engineering,
Mumbai, India.
poonambhogale@somaiya.edu

Shreyansh Kotak
Student,
Department of Computer Engineering,
K.J. Somaiya College of Engineering,
Mumbai, India.
shreyansh.kotak@somaiya.edu

Yash Pasar
Student,
Department of Computer Engineering,
K.J. Somaiya College of Engineering,
Mumbai, India.
yash.pasar@somaiya.edu

Abstract— **The abnormal uncontrolled cell growth in the brain, commonly known as a brain tumor, can lead to immense pressure on the various nerves and blood vessels causing irreversible harm to the body. Early detection of brain tumor is the key to avoid such compilations. Tumor detection can be done through various advanced Machine Learning and Image Processing algorithms. The various stages of brain tumor detection are image pre-processing, segmentation and feature extraction. Preprocessing includes enhancing the image by using various filters and removing noise. Segmentation includes methods like thresholding, region growing etc. Features like contrast, skewness, entropy are calculated for the extracted tumor. Different classifiers like Artificial Neural Network, Naïve Bayes are used to classify the tumor as benign or malignant.**

Keywords—Brain Tumor; Segmentation; MRI; Feature Extraction; Classification; Genetic Algorithm; Particle Swarm Optimization.

I. Introduction

The advancing technology has had a profound effect on the field of medical imaging. The building blocks of the body are the various cells, which make up the organs. A tumor is a disease arising out of these cells. Tumors can be either benign (primary) or malignant (metastatic or secondary). A benign tumor is static, continuing to grow in size, applying enormous pressure on the surrounding tissues of the brain causing problems. On the other hand, malignant tumors can originate somewhere else in the body as a mass of cancerous cells and migrate to the brain [1].

Expert opinion, Human Inspection and Biopsy are few of the methods that are available to diagnose tumors. Some of the drawbacks of these methods are excessive time consumption, inaccurate inspection etc. Hence, image-processing techniques are very helpful in the detection of brain tumors. Some of the current imaging techniques are - Computed Tomography, Positron Emission Tomography and Magnetic Resonance Imaging.

The most common imaging technique used to detect and clearly visualize the brain tumor formation, is the magnetic resonance imaging. It gives us a detailed analysis of the unaffected or healthy tissues of the brain and the affected tissues [2]. Computer aided diagnosis is based on the study of the various cerebral tissues of the brain – white matter, gray matter and cerebrospinal fluid. Careful analysis if these scans helps in planning an effective treatment for every patient.

The paper is organized to highlight the existing methodology in section II, a brief comparison of the two most efficient segmentation algorithms in section III, a survey of the various research papers in section IV, concluding in section V.

II. Survey of Existing Methodology

Existing methodology – segmentation and classification, have been surveyed and the results have been tabulated in Table I, given below. The methodology used in the past by various researchers for segmentation of the brain tumor and then extracting the features have some flaws which give low accuracy in the results. The most common segmentation methods used are K-means segmentation and the watershed segmentation, which lead to over segmentation of the tumor [3]. The commonly used classifier SVM has found to give inferior results as compared to the newer classification algorithms based on neural networks.

TABLE I. VARIOUS METHODOLOGIES STUDIED

Year	Method	Remark	Ref
2010 2012	Watershed Segmentation	A simplified algorithm, very accurate and has low computational overhead.	[9] [10] [11]
2012	Bayesian HMM, SVM	Using tumor probability map with SVM classifier or Hidden Markov Chains.	[12] [13]
2013	Neural Network based	Supervised and unsupervised learning, for tumor detection. Accuracy is dependent on feature selection and inputs given.	[14]
2013 2012	Support Vector Machine	SVM accuracy depends on the dataset and the application.	[15] [16] [17]
2015	K-means closeting, SVM	Better segmentation effect for Low SNR brain MR Images.	[18]
2016	Convolution Neural Network	More accurate results and reliable information for clinic treatments.	[19]

III. PROPOSED METHODOLOGY

The major steps followed in the detection of a tumor from a MRI Scan are as follows: Pre-processing, Segmentation, Feature Extraction and finally the classification. The flowchart in Figure 1 shows the major steps with snapshots of the output at each stage. The process begins with the accumulation of a clean data set of either T-1 weighted or T-2 weighted MRI scans.

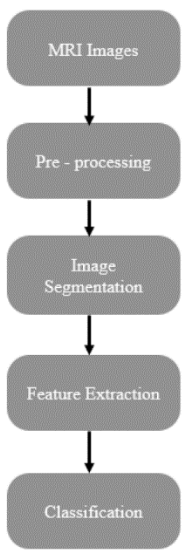
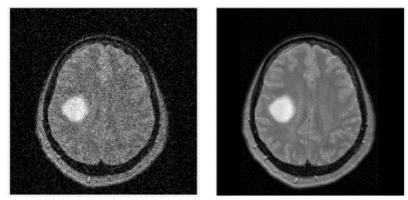


Figure 1: *Steps for Image Processing*

A. Preprocessing of the MRI Images

Preprocessing is the stage where noise is removed and any minute details are ameliorated. Clinical MRI scans riddled with noise reduce the accuracy of the segmentation algorithm output [3]. Multiple filters are used to eliminate the various kinds of noise. Median filters are used to eliminate salt and pepper noise while anisotropic filters are used to preserve the edges. To procure segmentation without any noise, this module is necessary. This refinement revamps the overall image quality. Figure 2 shows the contrast of a noise riddled image and a preprocessed image.



Original Image **Enhanced Image**

Figure 2: *Image Pre-Processing*

B. Segmentation Methods

1. Boundary Approach (Thresholding)

Thresholding is one of the most basic methods of segmentation that is used to isolate the tumor. All the pixels are allocated to a category based on the range that they lie in. That is for a certain threshold value t, the pixel located at position (i, j) with a grayscale value f_{ij} as shown in equation (1)

$$\text{Pixel (i, j)} = \begin{cases} 0, fij \leq t \\ 1, fij > t \end{cases} \quad (1)$$

2. Edge-Based Approach

In the edge based segmentation approach, the detected edges are assumed to be the representative boundaries of objects. Obtaining a closed distinct edge from this approach is highly unlikely. Additionally it is imperative to implement edge linking to join partial edges to gain a complete closed distinct edge of an object.

3. Region based Approach or Clustering

Region based segmentation or Clustering based segmentation is based on the pixel connectivity. A pixel can be four, six or eight connected in a 2-D image. It implies that all the pixels in a certain region are congruent to each other or have a somewhat similar value. The focus is more on finding pixels that satisfy the connectivity criteria than the edges of the object. Clusters are made up of congruent pixels. The various clustering algorithms used are:

a. K-means Algorithm:

An elegant solution to the clustering problem, the K-means algorithm is established on the theory that the centers of gravity of the various cluster elements represent the cluster [4], [5]. The algorithm clusters a dataset of 'n' objects (user input) into a set of k clusters (user input). It also computes the distance of the clusters belonging to the cluster centers of the objects.

b. Genetic Algorithm

John Holland founded the concept of genetic algorithms in the 1860's [5]. Holland's concept was advanced to generate a new population of chromosomes from the older population by taking a natural selection of two genetic operations – crossover and mutations. Genetic algorithms have three main stages: Recombination, Mutation and Selection. The selection operator selects the best individual chromosomes from a particular population. New chromosomes are created by crossing over the existing chromosomes among themselves. The new chromosomes are then transferred to the next population. Initially there are numerous possible solutions However, with every new generation, the number of viable solutions decreases until the most optimal, superior solution is obtained.

c. Particle Swarm Optimization

Particle swarm optimization is an optimization algorithm modelled after the simulation of social behavior of birds in a flock. It is a population-based algorithm. A group of random particles or solutions forms the initial solution of the algorithm. This initial solution then searches for the optima by subsequently updating the generations [6]. Every swarm particle traverses the search space constantly adjusting its position basis the distance from its own personal best position in the swarm and from that of the best particle of the swarm. A fitness function measures the fitness value of every particle in the swarm.

d. Jaya Optimization Algorithm

A newly implemented algorithm by Jaya Rao. The approach is built on the theory that the solution cluster resulting from the problem moves from the best-case option to the worst-case option basis the defined objective function (2)

$$X'_{j,k,i} = X_{j,k,i} + r1_{,j,i} \left(X_{j,best,i} - |X_{j,k,i}| \right) r2_{j,i} \left(X_{j,worst,i} - |X_{j,k,i}| \right) \quad (2)$$

m represents the design variables (j=1,2,3,4,… m) in the i^{th} iteration for 'n' candidate solutions. *Best* and *worst* represent the best and worst solution clusters in all the presented solution clusters. Here $X_{j,k,i}$ represents the j^{th} variable of the k^{th} solution for the i^{th} iteration. The absolute value of $| X_{j,k,i} |$ is used to further increase the searching capabilities [7].

C. Feature Extraction

The complexity of the brain makes the isolation of the tumor a difficult task. Various parameters are taken into consideration for feature extraction of the segmented tumors from the MRI scans. The main parameters for the area of interest are the expressions of the Gray-Level Co-Occurrence Matrix (GLCM) descriptors – Autocorrelation, Contrast, Correlation, Cluster Prominence, Cluster Shade, Dissimilarity, Energy, Entropy, Homogeneity and Maximum Probability [20]. The results of feature extraction are used to further classify the tumor as benign or malignant.

D. Classification

Classification is done basis the results of the feature extraction. Various patters are mapped according to the extracted features and then a classification is made for the extracted tumor. Various algorithms used for the classification are Artificial Neural Networks, Tree J84, Naïve Bayes and the Lazy IBK [8].

IV. COMPARISION OF SEGMENTATION ALGORITHMS

From the above surveyed methodology, theoretically, the best segmentation methods are Genetic Algorithms and Particle Swarm Optimization giving increased accuracy and efficiency as compared to the above described segmentation methods. Table II gives a brief comparison between the two methods. Theoretically, using either of these two method for segmentation of the tumor should give better efficiency and accuracy.

TABLE II. COMPARISION BETWEEN GENETIC ALGORITHMS AND PARTICLE SWARM OPTIMIZATION

Genetic Algorithms	Particle Swarm Optimization
Genetic algorithms operate on populations of strings coded to represent the parameter set.	Particle Swarm Optimization is initialized with a group of random generated solutions (particles) and optimizes it iteratively.
Genetic Algorithms can implements three main operations i.e. selection, crossover and mutation	Analogies exist in PSO though they do not label its operations like GA

V. CONCLUSION

This survey paper gives a brief overview brain tumor segmentation and classification techniques. A comprehensive analysis of the various stages of image processing are presented. Various segmentation methodologies are elaborated. It can be concluded that the algorithms and the parameters used in the proposed system are all meant to increase the efficiency of the system by achieving better results. The boundary approach and the edge based approach for segmentation are very common but the region growing approach gives better results. Features extracted by using GLCM method help to increase efficiency as minute details of tumor by using various features can be extracted. Accuracy and reliability are of utmost importance in tumor diagnosis, as a patient's life depends on the results predicted by the system. Thus, the proposed methodology helps in increasing the accuracy and obtaining the desired results.

References

[1] R. Mishra, "MRI based brain tumor detection using wavelet packet feature and artificial neural networks," *Proceedings of the International Conference and Workshop on Emerging Trends in Technology - ICWET 10*, 2010.

[2] Y. K. Dubey and M. M. Mushrif, "Segmentation of brain MR images using intuitionistic fuzzy clustering algorithm," *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing - ICVGIP 12*, 2012.

[3] S. M. K. Hasan, M. Ahmad, and S. D. Ghosh, "Perceptive Proposition of Combined Boosted Algorithm for Brain Tumor Segmentation," *Proceedings of the International Conference on Advances in Information Communication Technology & Computing - AICTC 16*, 2016.

[4] J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*. Amsterdam: Elsevier/Morgan Kaufmann, 2012.

[5] E. Küçükkülahli, P. Erdoğmuş, and K. Polat, "Brain MRI Segmentation based on Different Clustering Algorithms," *International Journal of Computer Applications*, vol. 155, no. 3, pp. 37–40, 2016.Mohsen, Fahd, et al. "A new image segmentation method based on particle swarm optimization." *Int. Arab J. Inf. Technol.* 9.5 (2012): 487-493.

[6] Mohsen, Fahd, et al. "A new image segmentation method based on particle swarm optimization." *Int. Arab J. Inf. Technol.* 9.5 (2012): 487-493.

[7] R. V. Rao, "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems," *International Journal of Industrial Engineering Computations*, pp. 18–34, 2016.

[8] M. Al-Ayyoub, G. Husari, O. Darwish, and A. Alabed-Alaziz, "Machine learning approach for brain tumor detection," *Proceedings of the 3rd International Conference on Information and Communication Systems - ICICS 12*, 2012.

[9] Q. Sun and H. Tian, "Interactive image segmentation using power watershed and active contour model," *2012 3rd IEEE International Conference on Network Infrastructure and Digital Content*, 2012.

[10] I. Maiti and M. Chakraborty, "A new method for brain tumor segmentation based on watershed and edge detection algorithms in HSV colour model," *2012 National Conference On Computing And Communication Systems*, 2012.

[11] Y. Liu and Q. Zhao, "An improved watershed algorithm based on multi-scale gradient and distance transformation," *2010 3rd International Congress on Image and Signal Processing*, 2010.

[12] Kaur, Mandhir, and Rinkesh Mittal. "Survey of Intelligent Methods for Brain Tumor Detection." *International Journal of Computer Science Issues (IJCSI)* 11.5 (2014): 108.

[13] K. Hu, G.-Y. Tang, D.-P. Xiong, and Q. Qiu, "A novel image segmentation algorithm based on Hidden Markov Random Field model and Finite Mixture Model parameter estimation," *2012 International Conference on Wavelet Analysis and Pattern Recognition*, 2012.

[14] D. Sridhar and I. V. M. Krishna, "Brain Tumor Classification using Discrete Cosine Transform and Probabilistic Neural Network," *2013 International Conference on Signal Processing , Image Processing & Pattern Recognition*, 2013.

[15] A. Ladgham, G. Torkhani, A. Sakly, and A. Mtibaa, "Modified support vector machines for MR brain images recognition," *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, 2013.

[16] P. Su, Z. Xue, L. Chi, J. Yang, and S. T. Wong, "Support vector machine (SVM) active learning for automated Glioblastoma segmentation," *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012.

[17] H. Li and Y. Fan, "Label propagation with robust initialization for brain tumor segmentation," *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012.

[18] J. Liu and L. Guo, "A New Brain MRI Image Segmentation Strategy Based on K-means Clustering and SVM," *2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2015.

[19] R. Lang, L. Zhao, and K. Jia, "Brain tumor image segmentation based on convolution neural network," *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2016.

[20] Akilandeswari, U., R. Nithya, and B. Santhi. "Review on feature extraction methods in pattern classification." *European Journal of Scientific Research* 71.2 (2012): 265-272.

Paper Presentation Certificate, INDIACom Conference:

Ref. No.:BV/ICAM/ND/ 821-21/2017-2018

16/03/2018

IEEE Advancing Technology for Humanity Delhi Section

CSI Delhi Chapter

IETE, Delhi Centre

ISTE, Delhi Section

Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi

12[th] INDIACom; 2018 5[th] International Conference on

Computing for Sustainable Global Development

INDIACom - 2018

IEEE Conference ID: 42835

(14[th] - 16[th] March, 2018)

This is to certify that Prof. / Dr. / Mr. / Ms. Apurva Mehta *of* K.J. Somaiya College of Engineering, Mumbai *has attended/contributed/presented a paper entitled* Brain Tumor Detection and Classification - A Survey *during INDIACom-2018; 12[th] INDIACom; 2018 5[th] International Conference on "Computing for Sustainable Global Development" organised by BVICAM, New Delhi.*

(Prof. A. K. Saini)
Conference Chair

(Prof. M. N. Hoda)
General Chair

Acknowledgements

The completion of our project has valuable contribution from various people with whom we interacted within the organization. This study, which is a complete fulfillment of the requirement for the degree of Bachelor of Engineering, would not have been possible without the constant support, mutual co-ordination of several people to whom we owe our sincere gratitude.

Firstly and most importantly, we would sincerely like to thank our mentor Prof. Poonam Bhogle. She has rendered her valuable knowledge of this subject and dedicated guidance with a touch of inspiration, vision and motivation. She has assisted us through any hurdles that we encountered by us giving plenty of early ideas, suggestions and modifications and encouraged us to complete the project to the best of our abilities.

We would like to thank our HOD Prof. Jyothi Rao who extended every facility to us for making and completing this project smoothly. We would also like to thank our fellow peers and friends who constantly challenge us while also inspiring us, and provide us with their selfless help and invaluable advices. Lastly, we thank the teaching and non – teaching staff of our college as well as the library staff for providing us timely essential information in the form of books and for their assistance in the laboratories.

Appendix

I. Minimum System Requirements

Software Requirements:

- MATLAB 2017b
- Online Database: www.cancerimagingarchive.net , database collected from pathology and radiology labs and BraTS Dataset from MIT.
- Microsoft Office

Hardware Requirements:

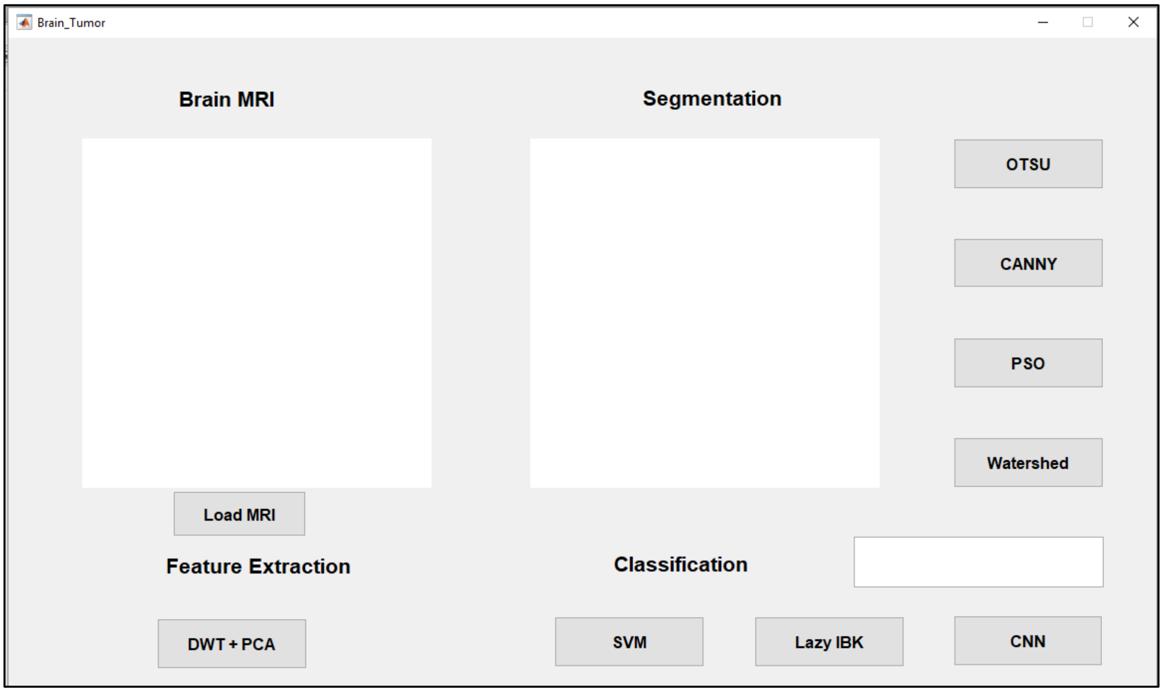
- Intel Dual Core Dual Processor or advanced version
- Minimum 8GB of RAM
- Minimum 1 GB of Hard disk Space

II. Technical Reference Manual

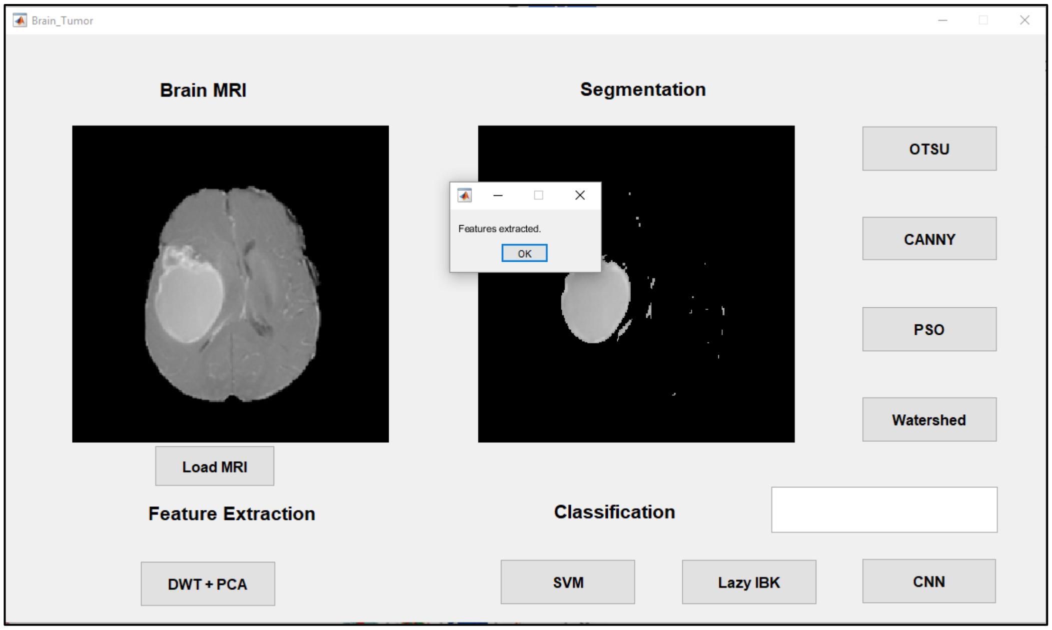
Installing MATLAB is requirement for implementation of this system. Licensed version of the software is obtained from the internet and next we have to follow the instructions given in their user manual.

All images loaded must be of desired format (BMP), and must not exceed 640*640 dimension.

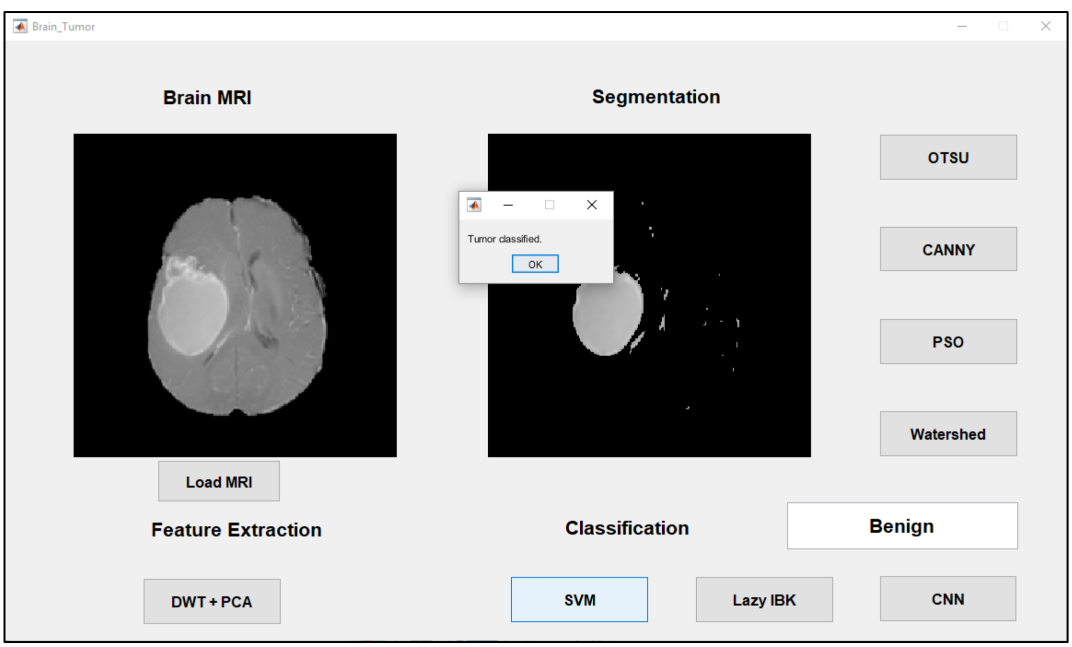
III. User Manual



1. Load folder containing tumor image training data folders with classified data to be used for training algorithm using Load MRI button below the white space for displaying the selected image.
2. Next, the user has to click the various buttons to the right, which signify the different segmentation algorithms. There are four options: Otsu, Canny, Watershed and PSO.
3. Once the user has clicked through the various buttons for segmentation, the segmented output will appear to the immediate left of the buttons. The user can then do a visual comparison of the original image and the segmented output.
4. User will now extract the features using the feature extraction button in the lower left corner of the GUI named ‘DWT+PCA’. The user can check the Excel generated for the extracted features. A pop-up alert lets the user know when the feature extraction is complete.



5. The next step is the classification. Similar to the segmentation step, the user selects the classification approach to be used. The three methods are: KNN, Lazy IBK and CNN. Once the user clicks the button a pop-up alerts the user that the image is classified.
6. Predicted group is displayed in the bottom right corner of the GUI.



The system is user-friendly. There is hardly any requirement of User Manual. Users just have to proceed with the software by clicking the buttons and the system will fulfill the user's requirements.

Plagiarism Report:

Blackbook + Appendix

by Apurva Mehta

Submission date: 28-May-2018 08:57PM (UTC+0530)
Submission ID: 969564511
File name: Blackbook_C2.pdf (2.61M)
Word count: 11334
Character count: 67819

Blackbook + Appendix

ORIGINALITY REPORT

14%	10%	2%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to K. J. Somaiya College of Engineering Vidyavihar, Mumbai Student Paper	3%
2	Submitted to Caledonian College of Engineering Student Paper	2%
3	Submitted to North West University Student Paper	2%
4	Submitted to Universiti Malaysia Perlis Student Paper	1%
5	unmeshasreeveni.blogspot.in Internet Source	1%
6	en.wikipedia.org Internet Source	1%
7	research.ijcaonline.org Internet Source	1%
8	www.xs4all.nl Internet Source	1%