INTELLIGENT INFORMATION DETECTION IN MEDICAL IMAGES

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

By

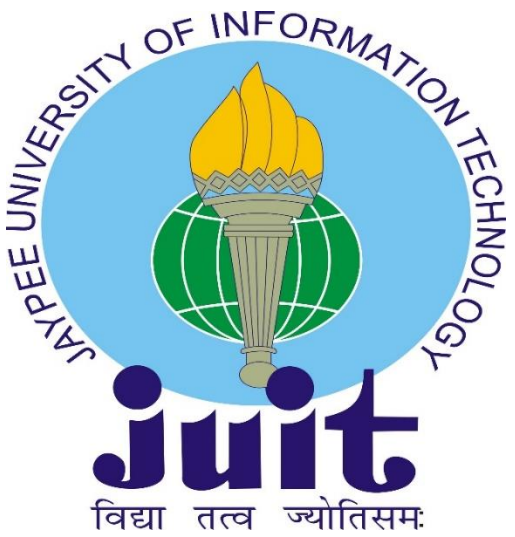
Gur Amrit Pal Singh (131288)

Under the supervision of

Dr. Pradeep Kumar Gupta

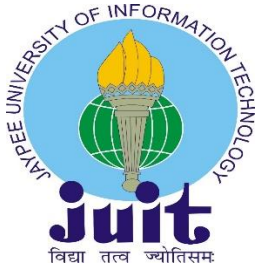
Assistant Professor (Senior Grade)

To



Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh



CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled **"INTELLIGENT INFORMATION DETECTION IN MEDICAL IMAGES"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the Department of Computer Science Engineering and Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2016 to May 2017 under the supervision of **Dr. Pradeep Kumar Gupta,** Assistant Professor (Senior Grade), Department of Computer Science & Engineering.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Gur Amrit Pal Singh, 131288.

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Pradeep Kumar Gupta
Assistant Professor (Senior Grade)
Department of Computer Science & Engineering
Jaypee University of Information Technology Waknaghat,
Solan-173234, Himachal Pradesh

Dated:

ACKNOWLEDGEMENT

I owe my profound gratitude to my project supervisor **Dr. Pradeep Kumar Gupta**, who took keen interest and guided me all along in my project work titled **— "INTELLIGENT INFORMATION DETECTION IN MEDICAL IMAGES"**, till the completion of my project by providing all the necessary information for developing the project. The project development helped me in research and I got to know a lot of new things in my domain. I am really thankful to him.

I would also like to thank my project partner **Rishabh Gupta** for his help and contributions, and **Dr. Ajay Deep Singh Sahni** for his continuous input and support on medical images.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

Angular Second Moment	ASM
Artificial Neural Network	ANN
Computed Tomography	CT
Computer Aided Detection / Diagnostics	CAD
Convolutional Neural Network	CNN
Decision Tree	DT
False Acceptance Rate	FAR
False Negative	FN
False Positive	FP
Fast Fourier Transform	FFT
Graphics Processing Unit	GPU
Gray Level Co-Occurrence Matrix	GLCM
Inverse Difference Moment	IDM
K Nearest Neighbors	KNN
Magnetic Resonance Imaging	MRI
Multi-Layer Perceptron	MLP
Positive Predictive Value	PPV
Radio Base Frequency	Rbf
Random Forest	RF
Rectified Linear Unit	ReLU
Red-Green-Blue	RGB
Region of Interest	ROI
Scale-Invariant Feature Transform	SIFT
Speed-Up Robust Features	SURF
Standard Deviation	SD
Stochastic Gradient Descent	SGD
Support Vector Machine	SVM
Three Dimensional	3D

True Acceptance Rate	TAR
True Negative	TN
True Positive	TP

LIST OF FIGURES

ABSTRACT

This Project report describes our objective of **INTELLIGENT INFORMATION DETECTION IN MEDICAL IMAGES**, where I tried to diagnose lung cancer using Digital CT scans by building a classifier. I have implemented supervised classifiers for cancer detection. Seven supervised learning classifiers implemented were K-Nearest Neighbors, Support Vector Machine, Decision Tree Multinomial Naïve Bayes, Stochastic Gradient Descent, Random Forrest, and Multi-Layer Perceptron. Images obtained from digital CT scans aren't always noise free, which could greatly hamper the detection process. In order to solve the problem of noise in images and to extract features from the given images, I have implemented various image processing techniques. These techniques varied from using Gaussian mean to remove noise from the images to using Otsu's thresholding method for converting the gray-scale images to binary images. Even though thresholding is quite a good method for conversion of gray-scale image into binary image, it still leaves behind gaps in the image which could mislead the classifier to wrongly classify an image. While these models were successfully implemented and evaluated on a small, segregated, consistent dataset; inconsistencies due to use of machine from different manufacturers, used on patients from different positions and angles in our dataset and in the real world application presented a real challenge in achieving our objective completely.

Chapter 1 – INTRODUCTION

1.1) Introduction: Image processing techniques in Magnetic Resonance Image (MRI), Computed Tomography (CT) Scans, X-ray, and Ultrasound diagnostics yield a great deal of information, which the radiologist has to analyses and evaluate in a very short amount of time. For diseases like cancer of any sort, this analysis has a critical need to be accurate. This makes the manual analysis very difficult and time consuming.

In simplest of terms, a cancer is any abnormal growth of body cells. A person cannot "Catch" cancer, it can't be transmitted from one person to another like common cold or flu. When any particular cell or a collection of cells is affected, then the growth of that particular cell or group of cells may become uncontrolled. There are many factors that can change the genetic coding of a cell in human body. Some of the factors that contribute to such a change are as chronic irritation, tobacco, smoke and dust, radioactive substances, age, sex, race and heredity, etc [1]. Cells that are unaffected grow in a well-regulated pattern which is governed by a person's genes which are transferred from both the parents to the child and is unique for each individual. When cancer sets in, a single cell or a bunch of cells abruptly starts multiplying in an uncontrolled and disorganized way, which if not stopped can lead to formation of lumps or tumors. Any tumor can be broadly classified into two categories – Benign and Malignant. A benign tumor never penetrates nearby cells where as a malignant tumor penetrates nearby cells, once a malignant tumor starts spreading it won't stop and can spread to other parts of the body also. Generally, benign tumors aren't considered dangerous but can turn dangerous if they develop vital structures required for their growth such as blood vessels or nerves. When this happens the tumor is said to be invasive.

If broadly classified, there are nineteen cancers that any healthy person can be affected with, along with many others. The one we worked on is Lung Cancer.

Lung cancer accounts for most number of cancer related deaths in both men and women worldwide. In United States alone it accounts for $12 billion in health care costs. Principle cause of lung cancer has been attributed to cigarette smoking which accounts to about 80 percent of total lung cancer cases worldwide. It may be noted that not everyone who is diagnosed with lung cancer is a smoker. Though most of the people with lung cancer are former smokers. Lung cancer in non-smokers can be caused due to a variety of reasons such as, exposure to radon, secondhand smoke, air pollution, or other factors. Workplace exposures to asbestos, diesel exhaust or certain other chemicals can also cause lung cancers

in some people who don't smoke.

About twenty-five percent of all people who have lung cancer may have no symptoms at all initially when the cancer is diagnosed. Lung cancer is usually identified when a chest X-ray or CT Scan is performed for any another reason. The rest of seventy-five percent of people may develop some sort of symptoms. These symptoms may arise due to direct effects of the primary tumor or due to effects of cancer that has spread to other parts of the body (metastases) or due to disturbances of hormones, blood, or other systems.

Main symptoms of lung cancer are cough, coughing up blood or rusty-colored phlegm, fatigue, unexplained weight loss, recurrent respiratory infections, hoarseness, new wheezing, and shortness of breath. Other symptoms may include:

- A new cough in a smoker or a former smoker.
- A cough that does not go away or gets worse over.
- Coughing up blood (hemoptysis).
- Pain in the chest area is a symptom in about twenty-fiver percent of people with lung cancer.
- Shortness of breath, resulting from a blockage in part of the lung, or collection of fluid around the lung (pleural effusion), or the spread of tumor through the lungs.
- Repeated respiratory infections, such as bronchitis or pneumonia [2].

In order to detect cancer radiologists, use various imaging techniques, but the one I am going to concentrate on is computed tomography (CT) Scans.

A computed tomography (CT) scan uses X-rays to produce comprehensive pictures of structures inside the human body [3].

1.2) Problem Statement: Progress in computer vision technology in recent years has allowed us to not only use medical imaging extensively but also to make use of data generated from the various medical imaging techniques, thus providing better diagnosis; improving not only treatment methods, but also lives of many patients as well as predication of many life threatening diseases in advance. Computer vision can exploit various properties of an image such as texture, shape, contour, and many more. Such information is vital for early detection of any diseases which in turn help doctors treat such disease in more effect way than previously possible.

1.3) Objectives: My vision is to build a Computer Aided Diagnostic (CAD) system that

can use computer vision to give a second opinion to the radiologist so that the process of diagnosis can be:

- Less time consuming
- Cost Effect

Our system would also:

- Reduce many errors that may arise due because of human negligence
- Increase the juxtaposition of results obtained from various laboratories.

Computer Aided Detection/Diagnostics (CAD) is a technology designed to reduce the observational oversights of physicians interpreting medical images and thereby decreasing the false negative rates. The fundamental aim of CAD is to increase the rate of detection of disease by reducing the false negative rate which are a result of observational oversights on the part of person examining these images. The use of a CAD instead of a second human observer has the advantage of not increasing the need of radiologists (or trained observers) [4].

I have used various machine learning algorithms to classify lung cancer from CT scans while having:

- High Accuracy rate of diagnosis (above 90%)
- High Precision (above 90%)
- High Recall (above 90%)

1.4) Methodology: There are two modules of my project.

A. **Feature Extraction:** First is the use of feature and texture based extraction methods to extract the cancerous areas from the given CT Scans. Using these techniques not only reduces the computations required to be performed in the second stage but also gives a clearer view of the problem lying ahead.
B. **Classifier:** Second module of my project is to design a classifier which can automatically classify a lung image as cancerous or not. For making this classifier we are using a seven machine learning algorithms which can then comparting accuracy of each one of them. Using both Euclidean distance based classifiers as well as back propagation classifier with multiple hidden layers.
C. **Testing:** Testing and training go hand in hand with classifier be modified and tested at the same time to improve the accuracy of the proposed system.

A separate data set called the test set was kept for final testing and accuracy measurement of the system. This was done so that the system does not learn from this data set, rather only classifies this data set. By doing this I would prevent the problem of overfitting the dataset. In order to completely remove the problem of overfitting, I also shuffled the training data set in each pass. The proposed system uses 80%, 70%, and 50% of the available data for training, and 20%, 30%, and 50% for testing respectively.

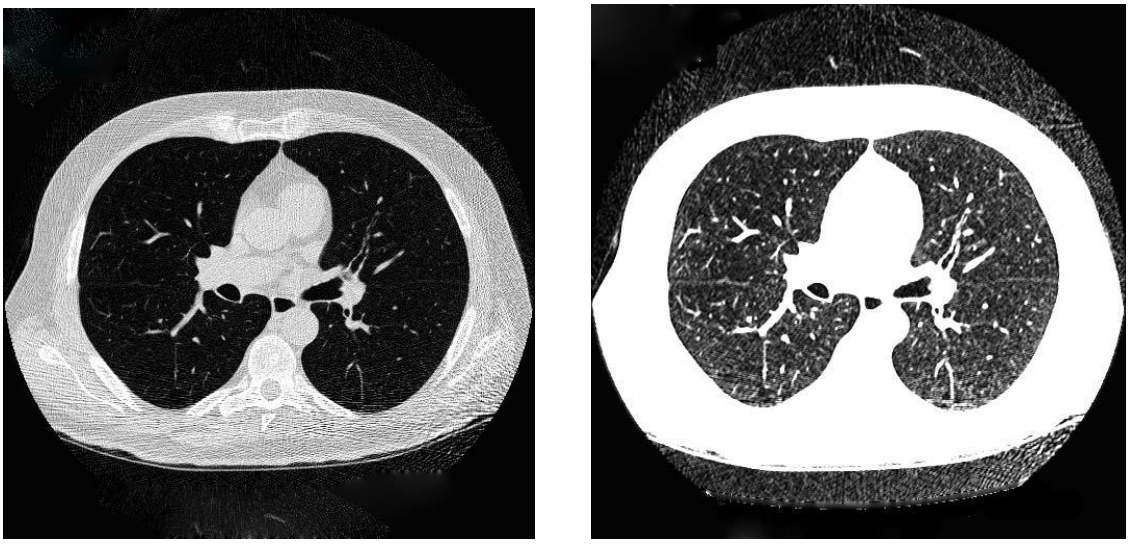
Chapter 2 - LITERATURE SURVEY

2.1 Introduction: Lung cancer is one of the leading causes of cancer related deaths in both men and women worldwide. It is estimated that more than 1.1 million people worldwide are diagnosed with lung cancer every year. This accounts to nearly 12% of the total number of people diagnosed with cancer every year. And about 1 million people each year die because of lung cancer, which accounts for more than 17% of the total number of cancer related deaths worldwide. Unlike skin cancer, lung cancer can't be seen with naked eye and its symptoms are often masked with other disease symptoms such as bronchitis, asthma, and coughing. As it is more common in cigarette smokers, detection of this disease becomes even harder. It is estimated that more than 82% of people suffering from lung cancer are diagnosed in much latter stages, that is when the symptoms become much severe and it becomes life threatening. Cancer cells may be carried away from the lungs where they form with the help of blood, or lymph fluid that surrounds lung tissue. Lung cancer usually spreads toward the center of the chest cavity, this is because the natural flow of lymph which is outwards from the lungs and inwards towards the center of the chest [5]. Metastasis is a condition which occurs when a cancer cell leaves the site where it was formed and moves into a lymph node or to another part of the body through the blood stream [6]. Computer Aided Diagnostics (CAD) can play a very important role in early detection of various life threatening disease by providing a second opinion to radiologists. In recent years many machine learning algorithms especially neural networks have been widely used for detection of lung cancer using medical images. Many of the algorithms proposed have achieved high accuracy rate [7].

2.2 Image Processing Techniques: In the studies [5], and [7] various image processing techniques were used for image enhancement and segmentation. The primary technique for image enhancement used was Gabor filter, and for image segmentation; thresholding, and morphological operations where suggested for improving the quality of images and for conversion of gray-scale images to binary images. These techniques have been explained below:

A. **Image Enhancement:** It is the process of improving the quality of an image from human perspective using various image processing techniques.

a) **Gabor Filter:** It is a linear filter. Gabor filter's impulse response is defined by multiplying Gaussian function with a harmonic function. Gabor filter has the property of multiplication-convolution (i.e., Convolution Theorem), due to which the Fourier Transform of the impulse response of a Gabor filter is the convolution of the Fourier Transform of the Gaussian function and Fourier Transform of the harmonic function. Figure 2.1 shows the effect of applying Gabor filter on the given input image [5].



a) Input Image *b) Output Image*

Figure 2.1. *Image enhancement technique using Gabor Filter*

B. Image Segmentation: Image Segmentation plays a very import role many of image analysis tasks. Most of the presently existing image recognition and description techniques depend greatly on the results of image segmentation. The crux of image segmentation is that it divides a given image into its corresponding objects or regions [7]. Segmentation in case of three dimensional (3D) medical images has a lot of useful applications for people belonging to medical profession, such as: volume estimation of the regions of interest, detection of abnormalities, classification, visualization, tissue quantification, and many more [8]. The aim of image segmentation is to represent an image into something that is simplified, is easier to analyze and provides more meaningful information. It is the process of giving label to each and every label pixel of the input image, so that the pixels which have some common characteristics share the same label. These characteristics can be color, intensity, or texture. Neighboring regions which have different labels, have significantly different characteristics.

Intensity of any pixel has two basic properties namely: similarity, and discontinuity. All image segmentation algorithms use one of these two intensity properties. Using similarity property, the image segmentation algorithms divide the image into regions that are similar based on some predefined criteria that is predefined, whereas, using discontinuity property the image is divided into regions based on abrupt changes in the intensity, such abrupt intensity changes can be found on the edges of an image. Example of image segmentation based of similarity property is histogram thresholding.

a. **Thresholding Approach:** Thresholding is one of the most commonly used image segmentation technique. There are various advantages of images obtained by using thresholding methods over gray scale images which contain 256 levels. Some of the advantages are: fast processing, smaller storage space and ease of manipulation. Thresholding being a non-linear operation converts a gray-scale image into a binary image (i.e., having only 2 levels). For a binary image only one level is assigned to each pixel based on whether the intensity of that pixel lies below or above a certain threshold value. This threshold value may be globally defined, or may be assigned using adaptive methods. Studies [5], and [9] have used Otsu's method for thresholding to converting a gray-scale image into a binary image. Otsu's method minimizes intra-class variance where it maximizes inter-class variance. Figure 2.2 shows the conversion of gray-scale CT Scan image of a lung containing cancer. Using Otsu's thresholding method this gray-scale image is converted into a binary image having only two levels (0, or 1) as shown in figure 2.3.

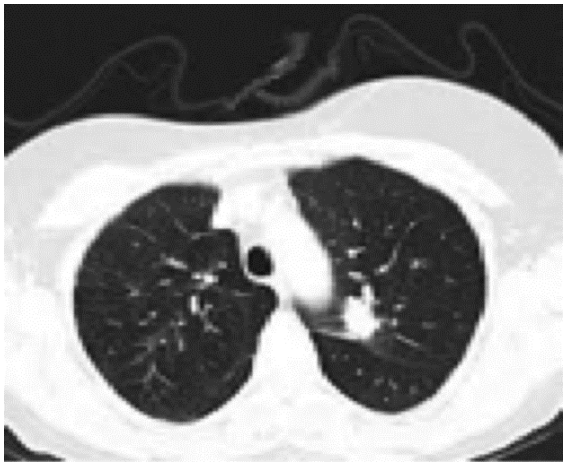


Figure 2.2. *CT Scan Image of a Lung with Cancer*

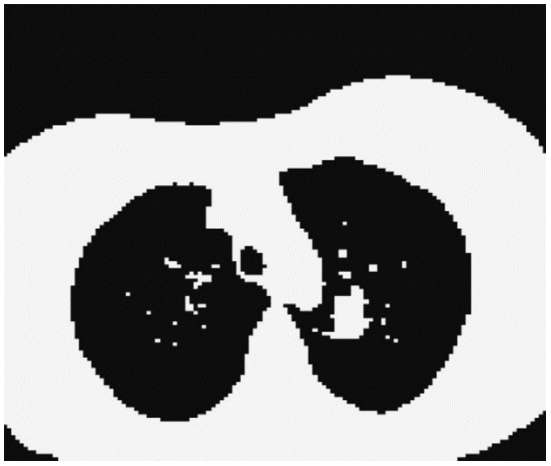


Figure 2.3. *Binary Image obtained after applying Otsu's Method*

Even after converting the gray-scale image into binary image some of the gaps are still left in the lungs which correspond to arteries or air present in the lungs as can been seen in figure 2.3. These gaps can cause the classifiers to predict wrong images as the give the illusion of a cancerous mass present. To solve this problem morphological operations must be performed to fill in the gaps left after applying thresholding method on the gray-scale CT scan images.

b. Morphological Operations: Once the input image is converted to binary image morphological operations are performed on it in order to fill in the gaps left by the conversion. Morphological transformations are some operations performed on an image based on the shape of the image. It is performed on binary images. These operations require two inputs, one is the original input image, and second one is called structuring element or kernel. Kernel decides the nature of operation to be performed. Two basic morphological operators are Erosion and Dilation. Then its variant forms like Opening, Closing, Gradient, etc, also comes into play. The morphological operation performed on the image is opening, which is erosion of an image followed by performing dilation on the eroded image. Figure 2.4 shows result of the morphological opening operation on the binary image, and figure 2.5 shows the final output image.

Figure 2.4. *Output of Morphological Opening Operation*

Figure 2.5. *Final Output Image*

2.3 Feature Extraction Techniques: Once the images have been preprocessed using image enhancement and image segmentation techniques, then features must be extracted from the final output image which are the supplied to classifiers for classification. Various feature extraction techniques have proposed by studies [9], [10], and [11]. These have been explained blow:

A. Mean: It is the average of all pixel intensity values. Mathematically mean is expressed as:

$$\mu = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} p\left(i,j\right) \qquad (1)$$

Where *p(i, j)* is the value of pixel intensity at the point *(i, j),* and M x N is the size of the image.

B. Standard Deviation: It is the estimation of mean square deviation of the gray pixel value *p(i, j)* from its mean value μ.

$$\sigma = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(p\left(i,j\right)-\mu\right)^2} \qquad 2)$$

C. Skewness: It characterizes the degree of asymmetry of a pixel distribution in the specified window around its mean. It is a pure number and it characterizes only the shape of the distribution.

$$S = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p\left(i,j\right)-\mu}{\sigma}\right]^3 \qquad (3)$$

D. Kurtosis: It measures the peakness or flatness of a distribution relative to a normal distribution

$$K = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p\left(i,j\right)-\mu}{\sigma}\right]^{4} \quad (4)$$

E. Fifth and Sixth Central Moment: The fifth and sixth central moments are given respectively:

$$FifthCentralMoment = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p\left(i,j\right)-\mu}{\sigma}\right]^{5} \quad (5)$$

$$SixthCentralMoment = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p\left(i,j\right)-\mu}{\sigma}\right]^{6} \quad (6)$$

F. Entropy: It measures the randomness of the elements of an image.

$$E = \frac{\sum_{i=0}^{M}\sum_{j=0}^{N}\left(p\left(i,j\right)^{2}\right)}{M \times N} \quad (7)$$

2.4 Image Classification: It is the task for classifying an image into various classes based on the labels of the input training data set. There are various machine learning algorithms which can be used for the task of image classification. These machine learning algorithms can be classified into two categories, namely: supervised and unsupervised learning algorithms. Seven of these supervised learning algorithms proposed by various authors have considered for this thesis. These are:

A. K-Nearest Neighbors Classifier: A traditional kNN classification algorithm works by deciding a class of an input image by searching for k images in the neighborhood of the input image in the training set which are similar to it [12]. In study [13] the authors have proposed a new kNN based classification method for image classification which is based on local feature generation using SIFT, or SURF. This new approach is divided into following two stages: all the local features are classified based on the local features of the training data; second the complete image is then classified based on the classification of each local feature.

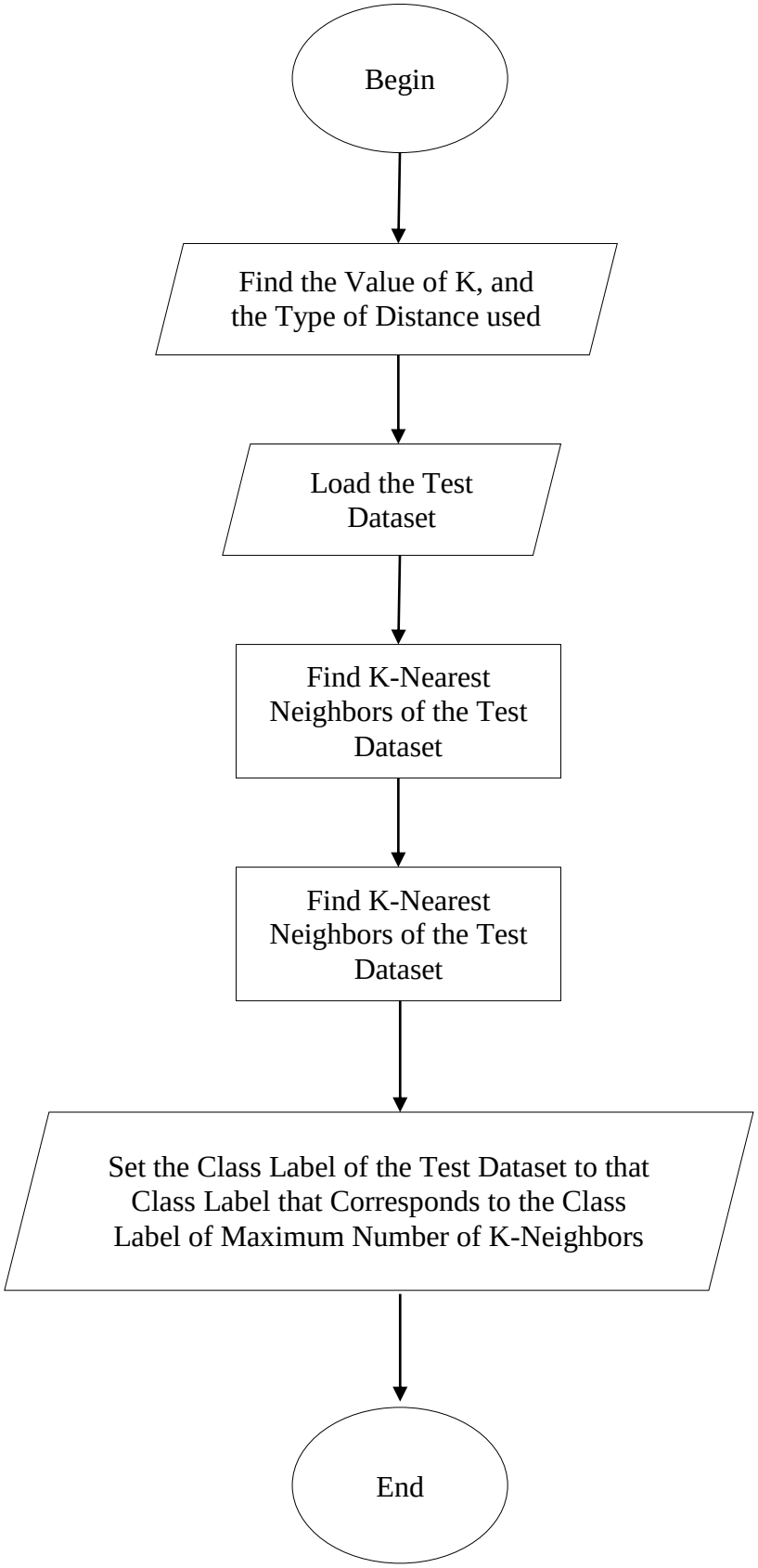


Figure 2.6. *Flow Chart of KNN Classifier.*

B. Decision Tree Classifier: Decision trees are conceptually simple to implement and have performed well different range of problems. Due to this simplicity of decision trees the number tree configurations are very large [14]. The flow chart of the mechanism of decision tree is shown in figure 2.7 [15].

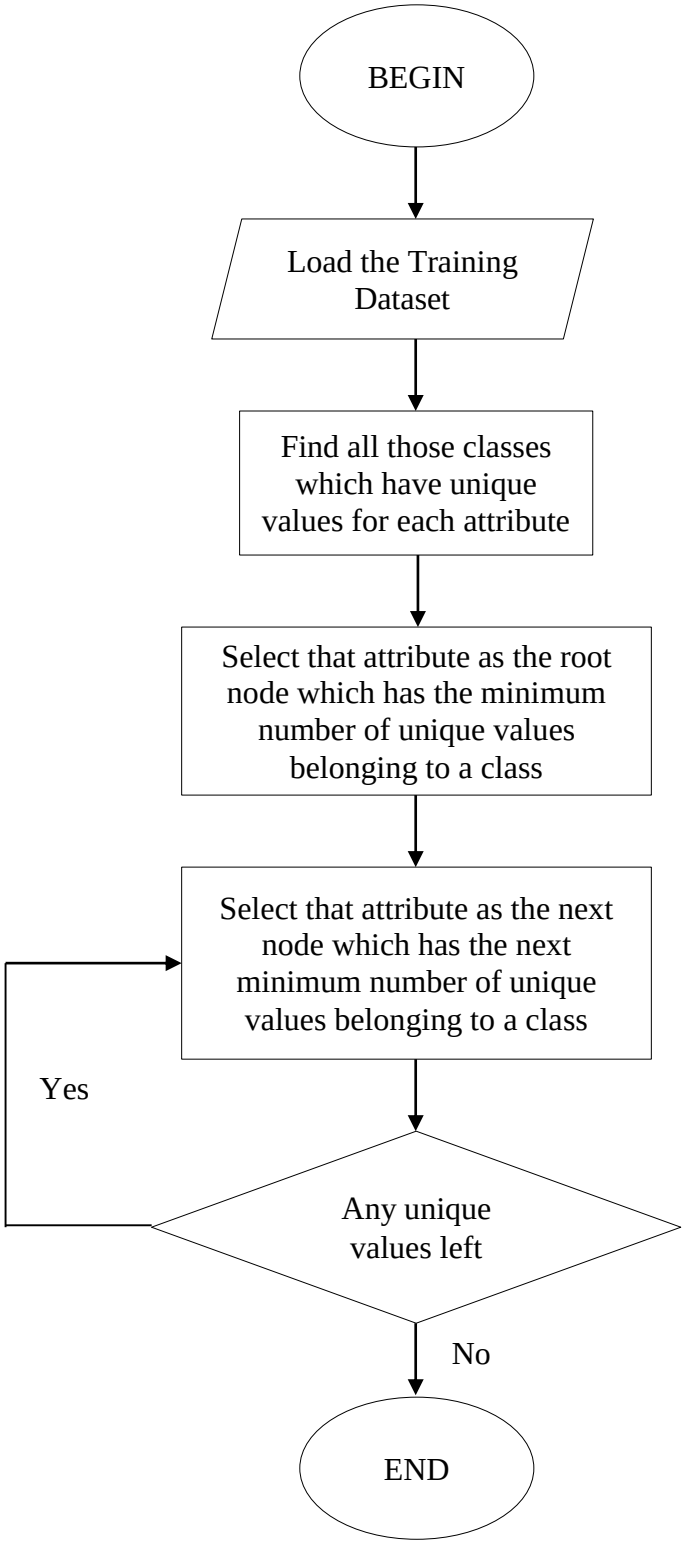


Figure 2.7. *Flow Chart of a Decision Tree Classifier*

C. **Support Vector Machine Classifier:** Though support vector machine has been around for past 22 years, it is still one of the major techniques for classification tasks. Initially applied only for the problem of character reorganization, SVM has come a long way from that and now can be easily applied for problems regarding image classification. A lot of research work has been done regarding the application of SVM for image classification [16].

D. **Naïve Bayes Classifier:** Naïve Bayes is a simple probabilistic based classifier. It is based on the assumption that all the features in the feature set are independent of each other. If the features are discrete than the best variant of Naïve Bayes classifier is either multinomial or Bernoulli distributions. On the other hand, when dealing with continuous distribution of features Gaussian distribution is the best proposed method. Study [17] showed that for normal text-classification tasks both multinomial Naïve Bayes and Gaussian Naïve Bayes classifiers performed significantly well and were better than classical Naïve Bayes classifier by a significant margin.

E. **Random Forest Classifier:** As already stated decision trees are simple machine learning algorithms, but suffer from the problem of configuration of different trees. A random forest classifier is a classifier that consists of a number of tree structured classifiers, and each of these tree structures independently casts its unique vote for the class label for an input. The class label which receives the most number of votes is selected by the random forest classifier as the class label of the input [18].

CHAPTER 3: SYSTEM DESIGN

This project involves the development of a classifier that operates on CT scans of the lungs to predict the cancer is present or not, and if it is present whether it is as benign or malignant.

3.1 Methodology: I have divided our project into five stages, namely: Image Acquisition, Image Preprocessing, Image Segmentation, Feature Extraction, Image Classification, Performance Evaluation. The flow chart of proposed system is shown below in figure 3.1.

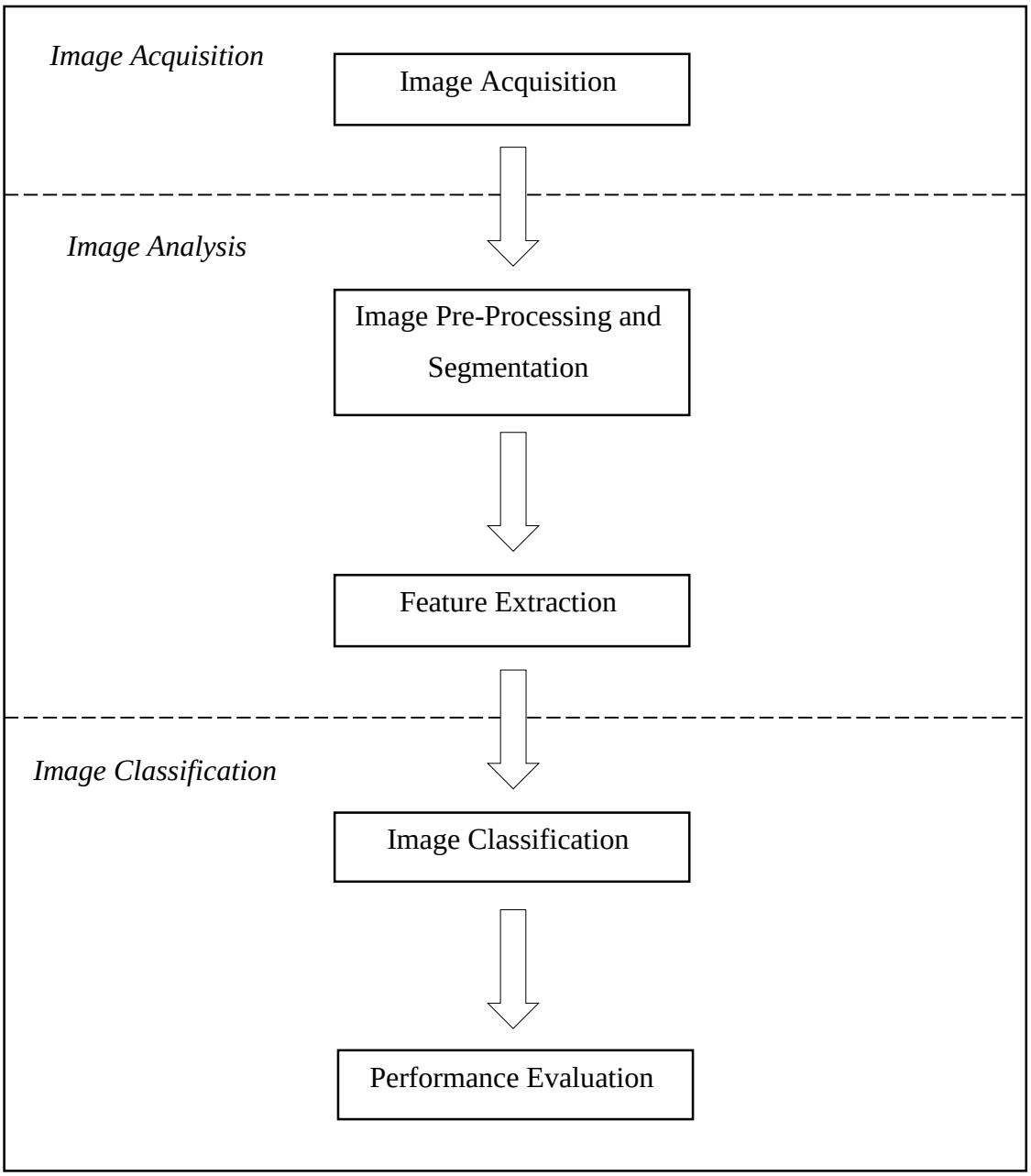


Figure 3.1. *The Flow Chart of Proposed System*

A. **Image Acquisition:** The first step towards the development of the project was acquiring data. For analysis, high power CT scan data was acquired online from [24], that is for benign and malignant. The Data set contained 21190 images of about 70 patients, with a total 37 cases of benign tumor, and 33 cases of malignant tumor.

B. **Image Analysis:** It is the process of working on images in order to improve the image quality for human readability and for removing noises from the images for efficient classification.

   a. **Image Pre-processing and Segmentation:** The first step of image analysis is image pre-processing. The acquired image is converted into a gray-scale image. Once the image is converted we apply thresholding methods for converting the gray-scale image into a binary image. A binary image consists of only two pixel values either '0', or '1', whereas in a gray-scale image each pixel can acquire any value between 0 and 255. In this research we have compared three thresholding methods, namely global thresholding, Otsu's Method, and Otsu's Method after applying Gaussian blur. When comparing these three thresholding methods we found out that Otsu's Method after applying Gaussian blur gives the best result. Figure 3.2 to figure 3.5 shows the effect of each these thresholding methods on our input image.

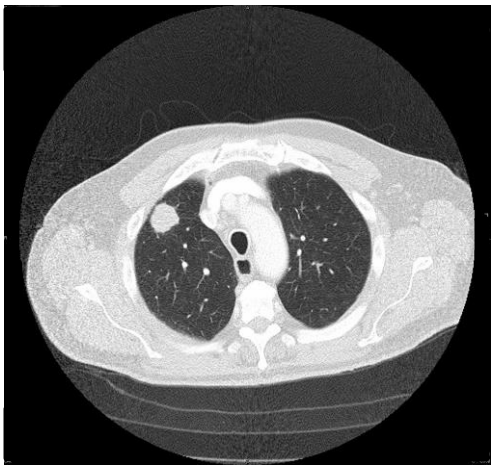


Figure 3.2. *Input Gray-Scale Image*

Figure 3.3. *Image after applying global Thresholding*

Figure 3.4. *Image after applying Otsu's Thresholding Method*

Figure 3.5. *Image after applying Gaussian Blur followed by Otsu's Thresholding Method*

After thresholding morphological opening operation is applied on the image to fill in the gaps left after thresholding. Morphological operations are some of the simple operations that can be performed on an image based on its shape. There are two basic morphological operations: Erosion, and Dilation, which are generally performed on binary images. Each of these morphological operations is implemented using a structuring element, also called as kernel. Just like soil erosion erodes the soil, erosion operation also erodes the boundaries of the foreground object. So a kernel slides over the image and a pixel value in the original image would be considered '1' of all the pixel values inside the kernel are '1', if it is not the case then the pixel value is set to '0'. On the other hand, dilation is opposite of erosion. A pixel value in the original image is considered to be '1' if one pixel value inside the kernel is '1'. So it increases the size of foreground object. Morphological opening operation is erosion followed by dilation [19]. Figure 3.7 shows morphological opening operation applied to the image after applying Gaussian Blur and Otsu's Thresholding Method.

Figure 3.5. *Image after applying Gaussian Blur followed by Otsu's Thresholding Method*

Figure 3.6. *Image after applying Morphological Opening Operation*

b. Feature Extraction: Texture refers to an object's appearance and the characteristics of its surface which is arises from the proportion of the elementary parts, arrangement, shape, density, and size of the object. Texture feature extraction is the collection such features through texture analysis process. In this research work we have considered texture feature extraction using two methods: 1) Using Gray-Level Co-occurrence Matrix (GLCM), and 2) Using Statistical Parameters.

GLCM can be applied to different texture feature analysis. It can be easily used for extracting second order texture information from images. The texture feature calculation which gives a measure of the variation in intensity at the pixel of interest can be calculated using the GLCM. Two parameters are used to compute the GLCM, these are; the relative distance between the pixel pair *d* measured in pixel number, and the relative orientation of the pixel pair *d*. The value of the parameter Θ is quantized in the following four directions (e.g., 0º, 45 º, 90 º and 135 º), many other permutations and combinations are also possible.

GLCM Features: GLCM has a total of fourteen different features but among them the most useful features are: contrast, dissimilarity, homogeneity, correlation, angular second moment (ASM), and energy are considered in the research. These five features are explained below:

Contrast: It is the measure of the intensity contrast between a pixel and its neighbor in the entire image [20]. It favors the contributions from *p(i, j)* away from the diagonal, i.e. $i \neq j$ [27]. Mathematically it is represented as:

$$CONTRAST = \sum_{n=0}^{G-1} n^2 \left\{ \sum_{i=1}^{G} \sum_{j=1}^{G} p(i,j) \right\}, |i-j| = n \quad (8)$$

Dissimilarity: It is defined as the sum of pixel values where *p(i,j)* is the absolute difference between *i*, and *j*. Mathematically it is represented as

$$DISSIMILARITY = \sum_{i=0}^{M} \sum_{j=0}^{N} p_{i,j} |i-j| \quad (9)$$

Homogeneity or Inverse Difference Moment (IDM): Mathematically it is expressed as:

$$HOMOGENEITY = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \frac{1}{1+(i-j)^2} p(i,j) \quad (10)$$

Inverse difference moment because of the denominator in the above equation (i.e., $(1+(i-j)2)^{-1}$) will receive small contributions from a homogeneous area *(i.e., where i ≠j).* This results in high value of IDM for homogeneous images and relatively low IDM value for inhomogeneous images.

Correlation: Correlation is a measure of gray level linear dependence between the pixels at the specified positions relative to each other. Mathematically it is expressed as:

$$CORRELATION = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \frac{\{i \times j\} \times p(i,j) - \{\mu_x \times \mu_y\}}{\sigma_x \times \sigma_y} \quad (11)$$

Angular Second Moment (ASM): It is the measure of homogeneity of an image. Any homogeneous scene contains very few gray levels, thus the corresponding GLCM will have only a few but high values of *p(i, j)*. Thus, the sum of squares will be high. Mathematically ASM is expressed as:

$$ASM = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} p(i,j))^2 \tag{12}$$

Energy: Mathematically energy is represented as:

$$ENERGY = \sqrt{ASM} \tag{13}$$

Statistical Features: From the region of interest we have extracted six statistical parameters namely: standard deviation, skewness, kurtosis, fifth and sixth central moments and mean. These are stated below:

Mean: It is the average of all pixel intensity values. Mathematically mean is expressed as:

$$\mu = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} p\,(i,j) \tag{14}$$

Where *p(i, j)* is the value of pixel intensity at the point *(i, j)*, and M x N is the size of the image.

Standard Deviation: It is the estimation of mean square deviation of the gray pixel value *p(i, j)* from its mean value μ.

$$\sigma = \sqrt{\frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (p\,(i,j) - \mu)^2} \tag{15}$$

Skewness: It characterizes the degree of asymmetry of a pixel distribution in the specified window around its mean. It is a pure number and it characterizes only the shape of the distribution.

$$S = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left[\frac{p\,(i,j) - \mu}{\sigma} \right]^3 \tag{16}$$

Kurtosis: It measures the peakness or flatness of a distribution relative to a normal distribution.

$$K = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p(i,j)-\mu}{\sigma}\right]^4 \quad (17)$$

Fifth and Sixth Central Moment: The fifth and sixth central moments are given respectively:

$$FifthCentralMoment = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p(i,j)-\mu}{\sigma}\right]^5 \quad (18)$$

$$SixthCentralMoment = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\frac{p(i,j)-\mu}{\sigma}\right]^6 \quad (19)$$

All the above mentioned 12 features every extracted for every patient in the dataset along with the class label for each patient was stored in a matrix format. After the image analysis step, image classification step was carried out using machine learning algorithms. Seven machine learning algorithms were employed for the classification task.

C. Image Classification: It is the task of classifying the input image into three class: '0' for non-cancerous, '1' for benign tumor, and '2' for malignant tumor. If the whole dataset available is used as the training set, the classifier will simply memorize the dataset and will give 100% accuracy on the dataset but as soon as the classifier sees a new image it will perform poorly. If we divide the dataset into training and testing set and continuously test on the testing dataset while tweaking the parameters, without reshuffling both the training and testing set, then the test set will bleed into the training set and is memorized by the classifier again which will result in over-fitting of data, hence the performance of the system on a completely new dataset would be poor. In order to address these problems what I did was to reshuffle the data set each time the classifier is trained and tested, this ensures that the classifier sees a new set of images each time it trains and tests, thus avoiding the problem of overfitting. Also I have tested my system with varying testing set size ranging from 0.20% of the total dataset size to 0.30% of the total dataset size, all the way to 0.50% of the total dataset size.

The seven machine learning algorithms have been used for the classification tasks are: K-Nearest Neighbors (KNN) Classifier, Support Vector Machine (SVM) Classifier, Decision Tree (DT) Classifier, Multinomial Naive Bayes Classifier, Stochastic Gradient Descent (SGD) Classifier, Random Forest Classifier, and Multi-Layer Perceptron (MLP) Classifier. Each one of them is explained in the following sections.

a. K-Nearest Neighbors (KNN) Classifier: KNN is an non parametric, supervised learning algorithm. Non parametric, means that it doesn't make any assumptions about the underlying data distribution. It assumes that the data is in a feature space or a metric space. Hence the data can be both a scaler as well as multi-dimensional vectors [21] Since the data points are in a metric space they are separated from which other by a distance. Though Euclidean distance is commonly used, need not necessarily be the case. The training data consists of a set of vectors and class labels assigned to each vector. In the simplest form these class labels are either '0', or '1', but KNN can work with any number of class labels. The K in the KNN is the number of neighbors that we need to consider for classifying a data point.

Let us assume that we have a point '*', and we need to assign it to either class 'circle' or class 'rectangle', as shown in figure 3.8 below. Using KNN, let us assume that K=3. Now with '*' as center we would draw circle so that it encloses only three data points on the given plane as K=3. As we can see that the closest three points to '*', calculated using Euclidean distance or any other method are all circles. Therefore, '*' is assigned to circle. The value of k determines the boundaries of each class [22].

Euclidean distance between two points *p,* and q in an n-dimension space can be calculated as [23]:

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_i - q_i)^2 + \ldots + (p_n - q_n)^2} \quad (20)$$

The value of 'k' for this research work has been selected as 5. KNN can also assign weights each class, thus a class having higher weights would be given priority. As for lung cancer classification, each of the three class have equal priority, hence uniform weights are assigned to each class. Other possible weights can be based on distance, that is inverse of distance, points

closest to the query point would have a greater influence than the points farther away from the query point. We can also define our own weights for each class [24].

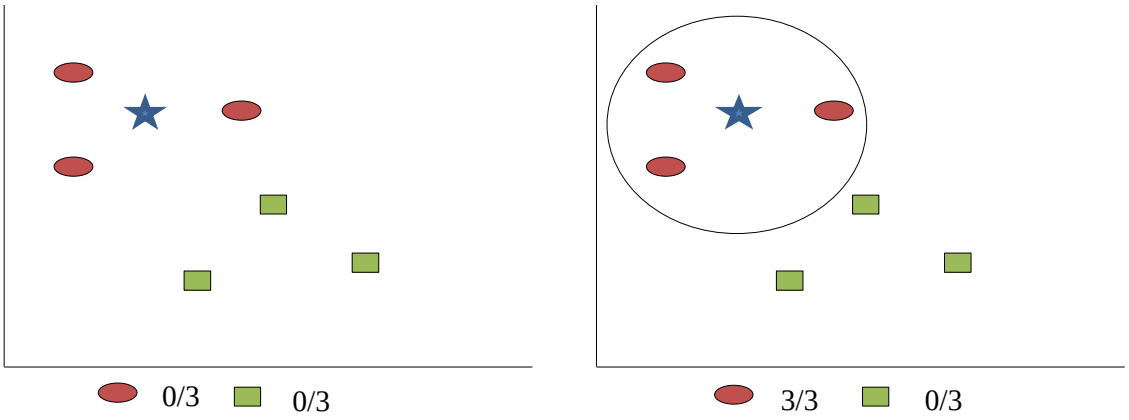


Figure 3.7. *Classification Using K-Nearest Neighbors (KNN) Classifier*

b. Support Vector Machine (SVM) Classifier: It is a binary supervised classifier, that it can classify two classes at most, though SVM can be extended for multi-class classification using kernel functions. Each data item in our data set is plotted as a point on an n-dimensional space, where n represents the number of features in our dataset, with each feature being a coordinate in that n-dimensional space. Classification is performed by finding the hyperplane which separates different classes. The hyperplane to be selected should be such that it maximizes the distance between different classes. An example of SVM is shown in figure 3.9 below.

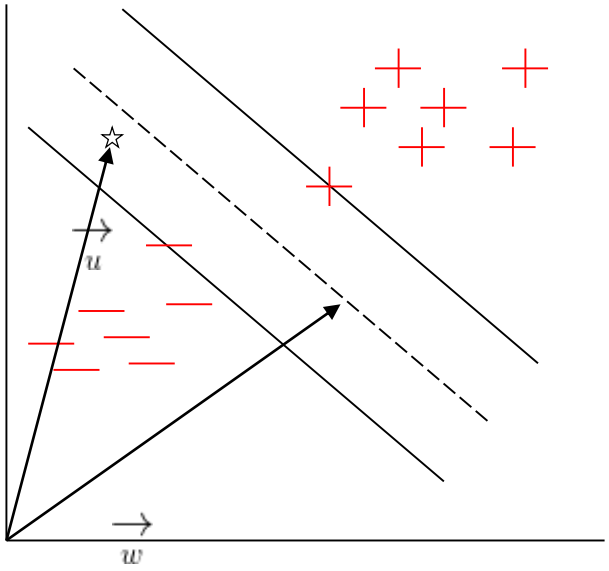


Figure 3.8. *SVM Hyperplane Example*

The dotted line drawn is the hyperplane separating the positive '+' and negative '-' examples in the feature space. This is drawn keeping in mid the wide street approach, i.e., maximum distance between the two examples. Consider a vector (say $\overrightarrow{w}$) perpendicular to the median line of the street (dotted line). Let's say that we have an unknown example '*' and a vector (say $\overrightarrow{u}$) pointing towards it. So what we are interested in knowing is the whether the unknown example lies on the left or right of the street.

Project the vector u down on the line perpendicular to the street (w).

$$w \cdot u \geqslant c \tag{21}$$

If the projection is big enough it will cross the median line of the street, then we can confidently say that '*' belongs to a positive sample.

$$w \cdot u + b \geqslant 0 \tag{22}$$

This is called the decision rule. Similarly if the projection is so small that it doesn't cross the median line of th street, then we can say that '*' belongs to the negative sample

Therefore,

$$w \cdot u + x_+ + b \geqslant 1 \tag{23}$$

Similarly,

$$w \cdot u + x_- + b \leqslant -1 \tag{24}$$

Now let us introduce y_i such that:

y_i = +1 (for positive examples), and (25)

y_i = -1 (for negative examples) (26)

Multiplying above two equations with y_i we get:

$$y_i (w \cdot x_i + b) \geqslant 1 \tag{27}$$

$$y_i (w \cdot x_i + b) \geqslant 1 \tag{28}$$

$$\boxed{y_i (w \cdot x_i + b) - 1 \geqslant 0} \tag{29}$$

For x_i that lie on non-dashed line the above equation is modified as:

$$\boxed{y_i (w \cdot x_i + b) - 1 = 0} \tag{30}$$

The space for an SVM is a convex space, this means that it will never get stuck in a local maximum, or minimum.

For samples which are not linearly separeable, SVM provides the kernel function, which transforms the space. The various kernel functions for SVM are linear, polynomial, radio base frequency (rbf), and sigmoid. Besides these custom made kernel functions can also be used. For this research I have used radio base frequency (rbf) kernel function, and a tolerance value of 0.001.

c. **Decision Tree (DT) Classifier:** It is a non-parametric supervised learning algorithm used for building classification models in the form of a tree structure. It divides the dataset into smaller subsets, while doing that it develops an associated decision tree. This results in a tree structure contacting decision nodes and leaf nodes. A decision node is the one with two or more branches, whereas leaf nodes are used for classification or decision making. The top node in the decision tree is called the root node. A decision tree is built in a top down manner, that is beginning from the root node towards the leaf nodes. Data is partitioned in subsets, and each subset contain similar value instances that is each subset contains homogenous data. Homogeneity of the data is calculated using entropy, which is the measure of randomness. If a subset is completely homogenous, then its entropy would be zero, however if the subset can be equally divided then its entropy is said to be one [19].

   After a subset is divided on the basis of attributes its entropy is decreased. This decrease in entropy is related to what is called information gain. Decision tree is constructed by finding those attributes that return the highest information gain (i.e., those branches that are most homogenous).

   Constructing a Decision Tree:

   Step 1: Calculate the entropy of the root node i.e., entropy of the whole dataset.

   Step 2: Whole dataset is divided into branches based on different attributes. Entropy of each divided branch is calculated and is added, so as to calculate the total entropy of the split. For acquiring information gain, the entropy obtained after spilt is subtracted from the entropy before the spilt, this should be positive, indicating decrease in entropy.

Step 3: Those attributes which provide the largest information gain are selected as decision nodes.

Step 4: Steps 1, 2, and 3 are repeated for all branches, till a branch with zero entropy is achieved. Such a branch forms the leaf node.

Step 5: The algorithm keeps on running recursively on all non-leaf nodes till all the data is classified.

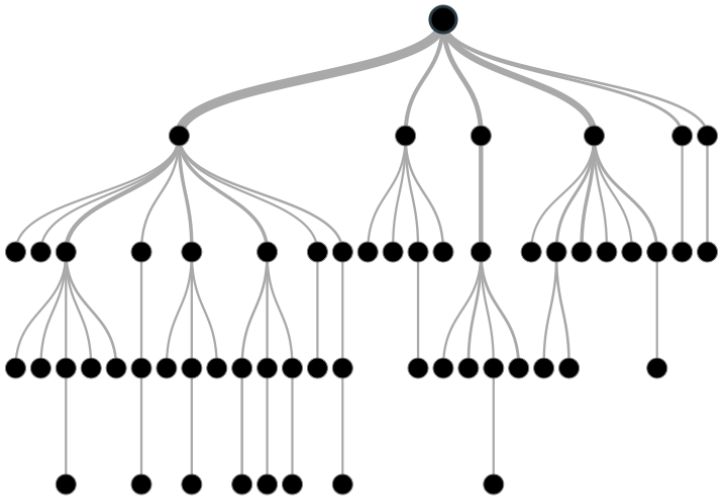


Figure 3.9. *Representation of a Decision Tree*

Deeper is the decision tree, more complex are the decision rules. Some of the advantages of using decision trees are:

- **i.** Easy to understand and implement.
- **ii.** Little data pre-processing is required as compared to other machine learning algorithms.
- **iii.** The overall cost of predicting data is logarithmic.
- **iv.** It can handle different types of data, both categorical and numerical.
- **v.** It can handle problems with multiple outputs.

Some of the major disadvantages of decision trees are:

- **i.** If more complex trees are created, it can lead to the problem of data over-fitting.
- **ii.** These are susceptible to changes in the data. Any changes in the data might lead to construction of completely different tree, thus making it unstable.

iii. It can lead to construction of biased trees if some of the classes dominate over the other. Hence balancing of data is must before applying decision tree classifier.

d. Multinomial Naive Bayes Classifier: Naive Bayes classifier works by using Bayes theorem to predict the classes of unknown datasets [26]. The basic assumption behind Naïve Bayes classifier is that the presence of a feature in a class are not related to presence of any other feature in that class. Thus, all the features contribute independently to the overall probability of the class. If '*y*' is a class variable and x_1 to x_n represent a dependent feature vector, then according to Bayes' theorem:

$$P\left(y \mid x_1, \ldots, x_n\right) = \frac{P\left(y\right) P\left(x_1, \ldots x_n \mid y\right)}{P\left(x_1, \ldots, x_n\right)} \quad (31)$$

This relation can be further simplified using the naïve independence assumption, that is:

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y), \quad (32)$$

Simplified relation:

$$P\left(y \mid x_1, \ldots, x_n\right) = \frac{P\left(y\right) \prod_{i=1}^{n} P\left(x_i \mid y\right)}{P\left(x_1, \ldots, x_n\right)} \quad (33)$$

It is given that $P(x_1, \ldots, x_n)$ is a constant, therefore for classification we use the following rule

$$P\left(y \mid x_1, \ldots, x_n\right) \propto P\left(y\right) \prod_{i=1}^{n} P\left(x_i \mid y\right) \quad (34)$$

Despite being over simple, naïve Bayes classifiers are very fast as compared to other more sophisticated algorithms. They can be trained using a small amount of data.

For this research work I have implemented multinomial naïve Bayes algorithm. Using multinomial naïve Bayes algorithm, we can implement the basic naïve Bayes algorithm for data that is distributed multinomially.

e. **Stochastic Gradient Descent (SGD) Classifier:** It is one of the simplest machine learning algorithms which has been around for a quite a long time. It can be easily applied to machine learning problems which are sparse and large-scale [27]. The major advantages of SGD are:

   i. Easy to implement.
   ii. High efficiency.

   Even though SGD is easy to implement it still has few loopholes such as:

   i. The number of hyperparameters required by SGD is quite large.
   ii. Feature scaling can greatly affect the results of SGD.

   In this research we have implemented SGD with learning rate *'α'*=0.0001, number of jobs *'n'*=-1, that is, all CPU cores are used to run SGD [28].

f. **Random Forest Classifier:** Random Forest classifier is an advance version of the decision tree classifier discussed earlier. It uses multiple simple classification trees to classify a new object. As the input vector is given to the Random Forest classifier, it uses each of the trees in the forest to classify the input vector. The classifier the chooses that classification which most of the trees generate. Every tree in the Random Forest classifier grows to the maximum extent possible. Every tree in the classifier has its own error rate. A tree which has a low error rate is said to be a strong classifier, so greater the number of trees which are strong classifiers, greater is the overall strength of the forest, thus lower is the error rate of the forest. Some of the major features of Random Forest classifier are [29]:

   i. It can easily run on large data sets.
   ii. It can give an estimate of important variables for classification.
   iii. It can handle missing data.
   iv. It can also be used for unsupervised clustering.
   v. It can handle datasets with large number of features and features.
   vi. It does not over fit the data.
   vii. It is quite fast.

   For this research I have used a Random Forest classifier in which the forest consisted of 10 trees. The trees are spilt till all the leaf nodes are pure, and also the minimum number of samples which are required to spilt an internal

nose are 2. Also the minimum number of samples for a node to be the leaf node is 1. The number of jobs *'n'*=-1, that is, all CPU cores are used to run the Random Forest Classifier [30].

g. Multi-layer Perceptron (MLP) Classifier: A multi-layer perceptron (or Artificial Neural Network – ANN) is a type of supervised learning algorithm with the learning function – $f(\cdot) : R^i \rightarrow R^0$, which learns on the training dataset, where *'i'* and 'o' are the dimensions of the input and output matrices respectively. Feature set is $X = x_1, x_2, ..., x_m$ and the label is *y*. The main difference between an MLP and logistic regression is that in between the input and the output layer, there can be *n* number of non-linear layers which are called the hidden-layer. Though an MLP can have many layers, the figure 3.11, shows MLP with only one hidden layer. The layer on the left side is called the input layer and contains the input feature set $\{x_i \mid x_1, x_2, ..., x_m\}$. The second layer is called the hidden-layer, and each neuron in this layer receives the input from the previous layer and transforms those values using a weighted linear summation, which then followed by an activation function (non-linear) $g(\cdot) : R \rightarrow R$. The final layer is the output layer which receives values from the hidden layers and converts those values into the final output values. Some of the major advantages of MLP are:

- **i.** It can even learn non-linear models, that is models which can't be separated using $y = mx + c$ line.
- **ii.** It can learn model in real time.

There are some of disadvantages of MLP, these are:

- **i.** It is susceptible to local minima if the hidden layers have a non-convex loss function. To solve this problem random weights are assigned.
- **ii.** It is also susceptible to feature scaling [31].

In this research I have implemented MLP classifier using backpropagation. The number of hidden layers used were 5. The activation function used was rectified linear rectifier unit function (or ReLU). For weight optimization *'sgd'* solver was used which is a stochastic gradient decent

optimizer. The learning rate *'α'* was initialized as 0.0001 and was set as adaptive to which keeps the learning rate constant till the time the training loss is decreasing. Once the training loss begins to rise for two consecutive epochs then the current learning rate, that is learning rate during the time training loss is increasing, is divided by five [32].

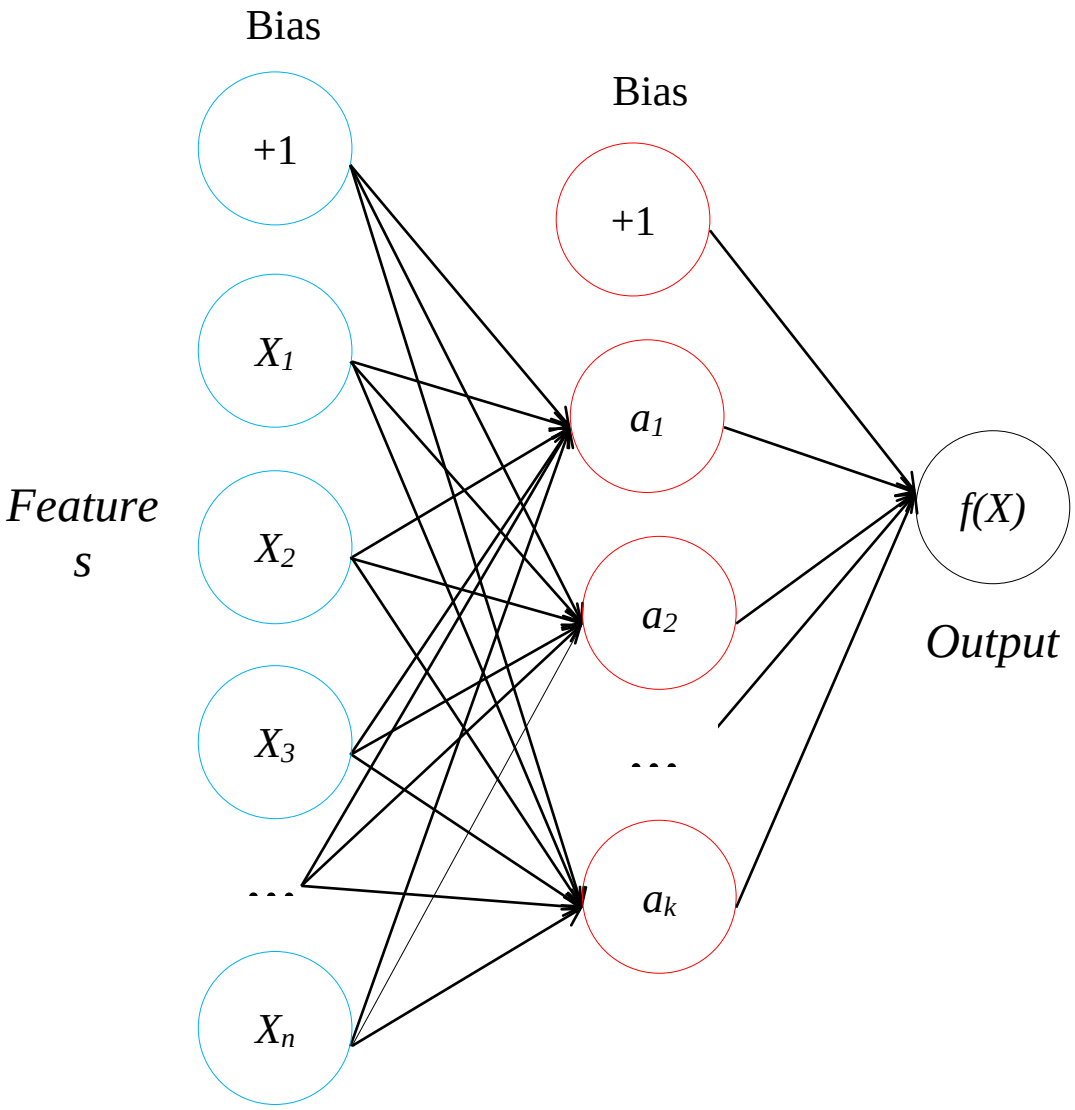


Figure 3.10- *MLP with only One Hidden-Layer*

After extracting all the GLCM features along the statistical features, those features were stored in a matrix format along with the class labels for each class. These class labels help identify whether that particular lung image contains cancer (class label '1' for benign, and

‘2’ for malignant) or not class label (class label ‘0’). These extracted features along with the class labels formed our training set. We then divided our training set in batches of three, first batch containing half of the total number of images, second batch containing about three-fourth of the total number if images, and the final batch containing full set of images. These images batches were nothing but the extracted features along with their respective class labels in a matrix format stored in *.txt* files. Each of these three *.txt* file was given to all the seven machine learning algorithms, and testing set was taken out from these files only using the cross–validation method. All these seven machine learning algorithms mentioned above were evaluated on following three parameters on all the three batches:

a. **Accuracy:** It is a statistical measure of how well a classifier correctly identifies or excludes a condition. The accuracy is the percentage of true results (both true positive and true negative) in the given data set.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (35)$$

b. **Precision:** Also called positive predictive value, it measures the total number of positive cases which the algorithm identifies.

$$Precision = \frac{TP}{TP + FP} \quad (36)$$

c. **Recall:** Also called as sensitivity, it measures the percentage of actual positive cases which the algorithm correctly identifies. That is the percentage of the images containing a cancerous nodule correctly classified by the algorithm as cancerous.

$$Recall = \frac{TP}{TP + FN} \quad (37)$$

Where;
True Positive (TP): Images containing cancer, classified as cancerous.

False Negative (FN): Images containing cancer, classified as non-cancerous.

True Negative (TN): Images not containing cancer classified as non-cancerous.

False Positive (FP): Images not containing cancer classified as cancerous.

CHAPTER 4: PERFORMANCE ANALYSIS

As stated in the previous section, I have implemented seven different classifiers. In this section, I will demonstrate how well they performed.

The dataset used included 512 X 512 images, categorized over 3 classes - non-cancerous, benign, and malignant.

All above mentioned seven models have been tested for varying training sizes and their test set accuracy, precision, and recall has been calculated and compared in the form of graphs plotted below.

4.1 Constant Test Set Size of 0.2 Percent:

Figures 4.1 to 4.9 show the accuracy, precision, and recall achieved using all the seven classifier when the size of data set was varied according to half of total dataset size, three-fourth of total data set size, and full dataset. The test set size for all three sizes of datasets was fixed at 0.2 percent of the total dataset size used.

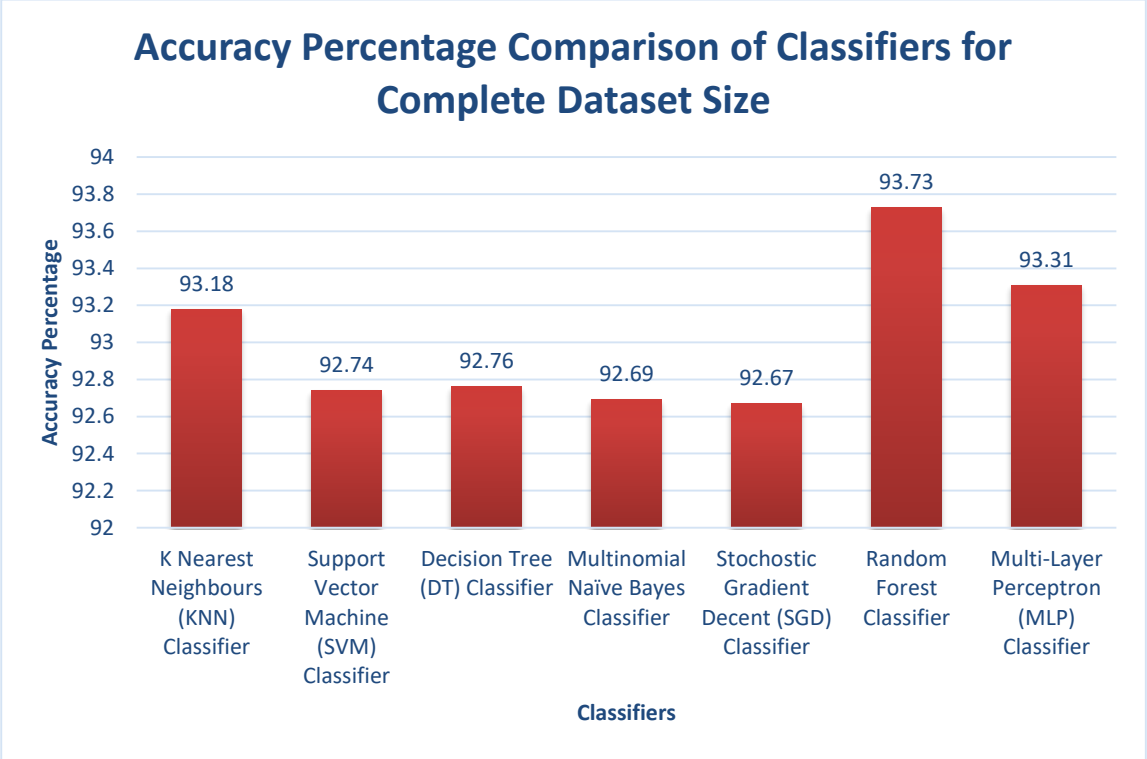


Figure 4.1. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.2 percent) and Complete Dataset Size (21190 Images)*

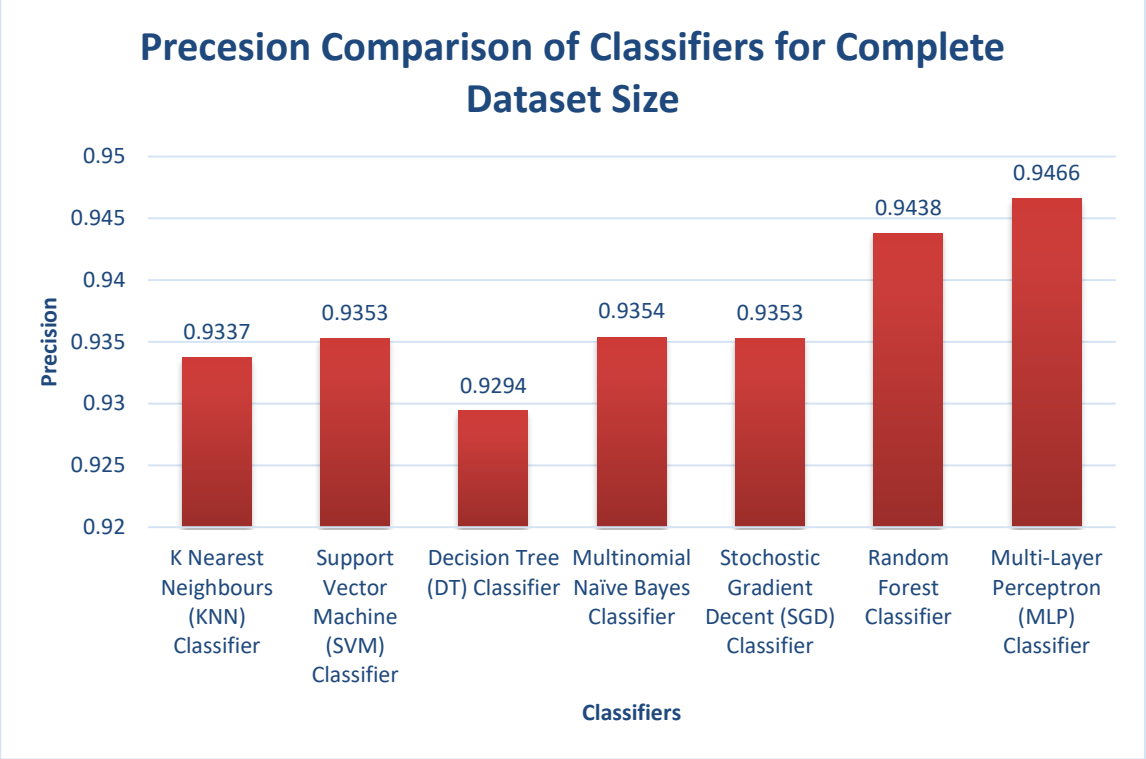


Figure 4.2. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.2 percent) and Complete Dataset Size (21190 Images)*

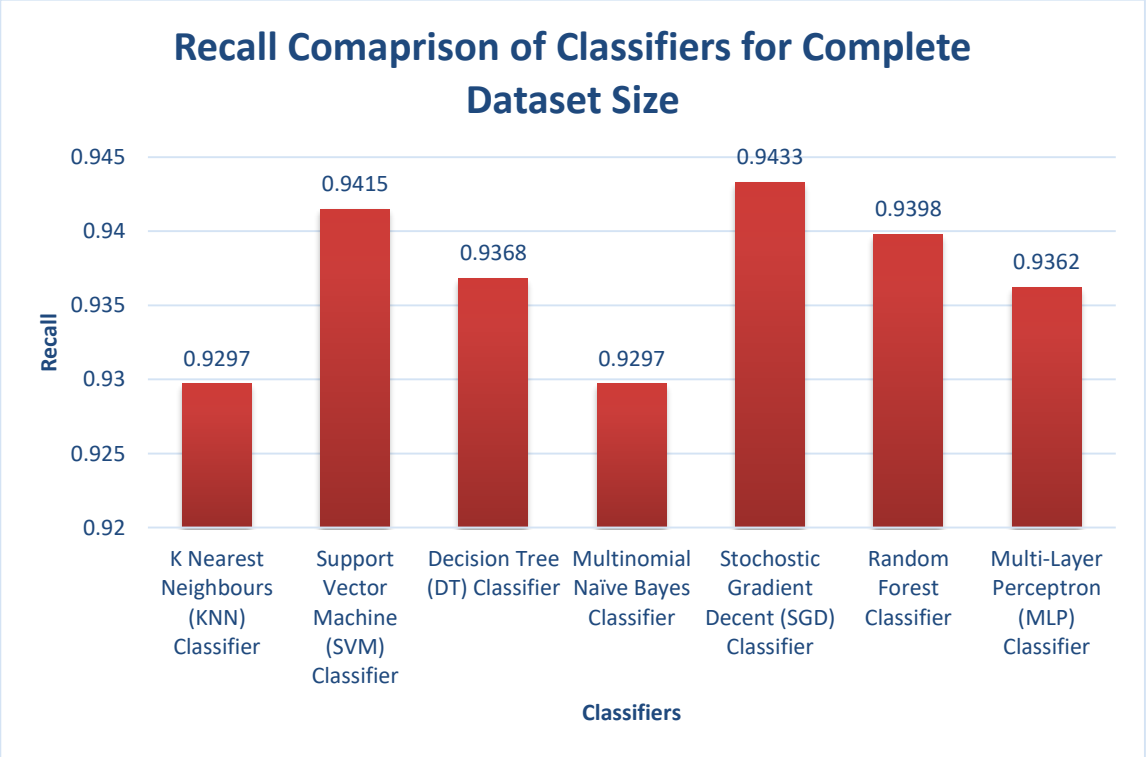


Figure 4.3. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.2 percent) and Complete Dataset Size (21190 Images)*

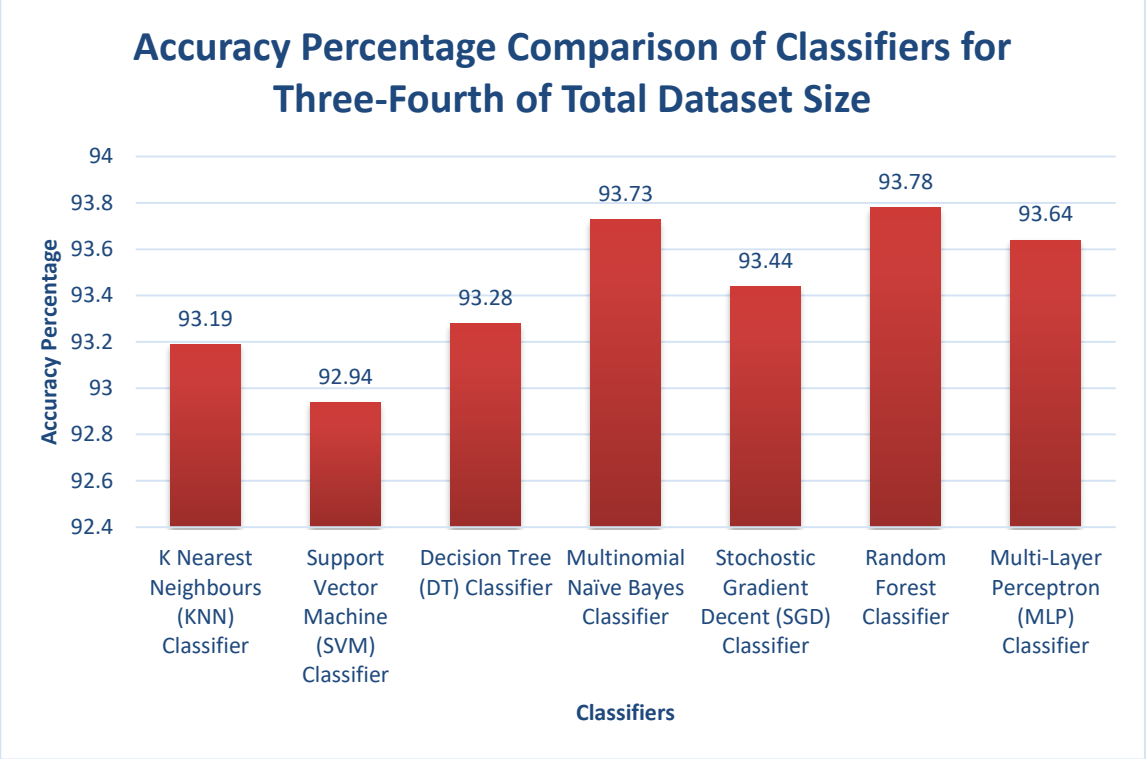


Figure 4.4. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.2 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

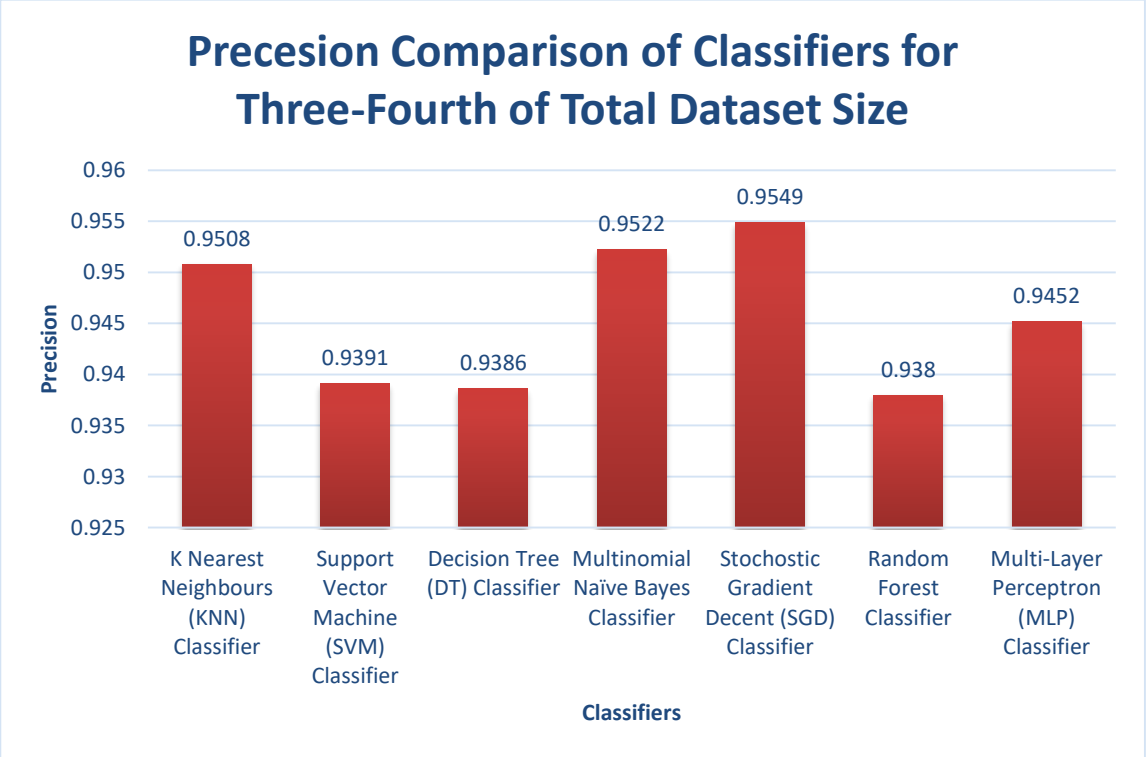


Figure 4.5. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.2 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

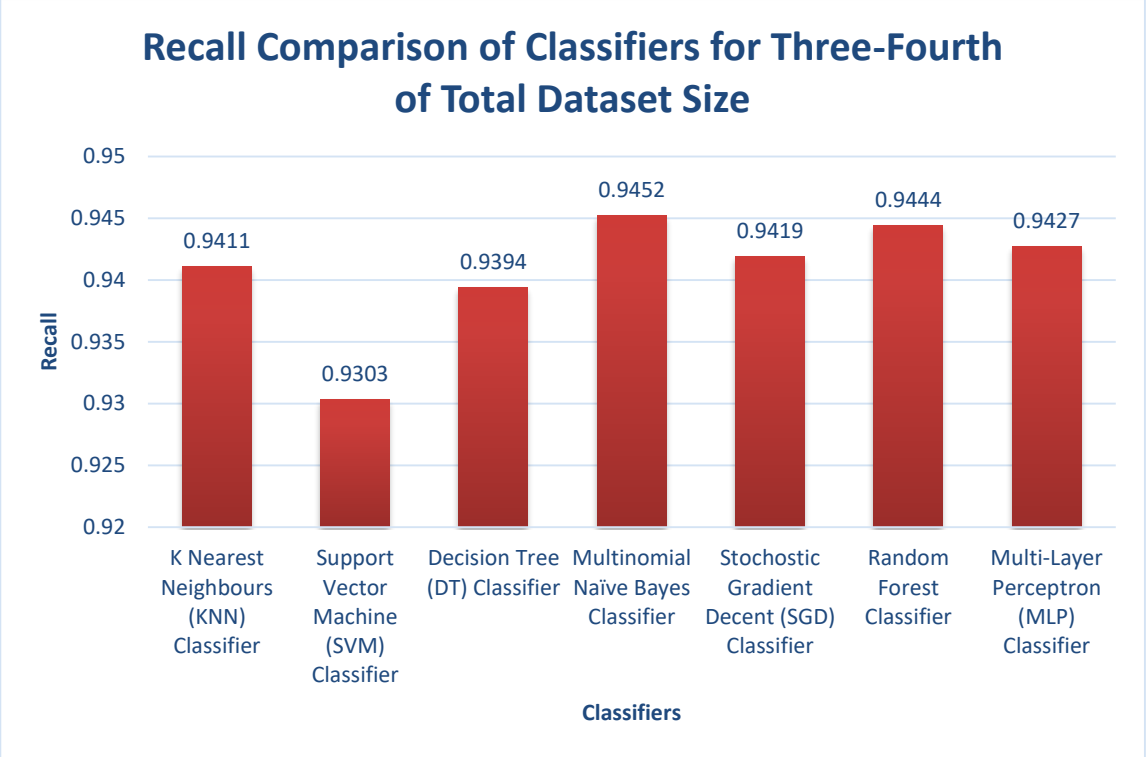


Figure 4.6. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.2 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

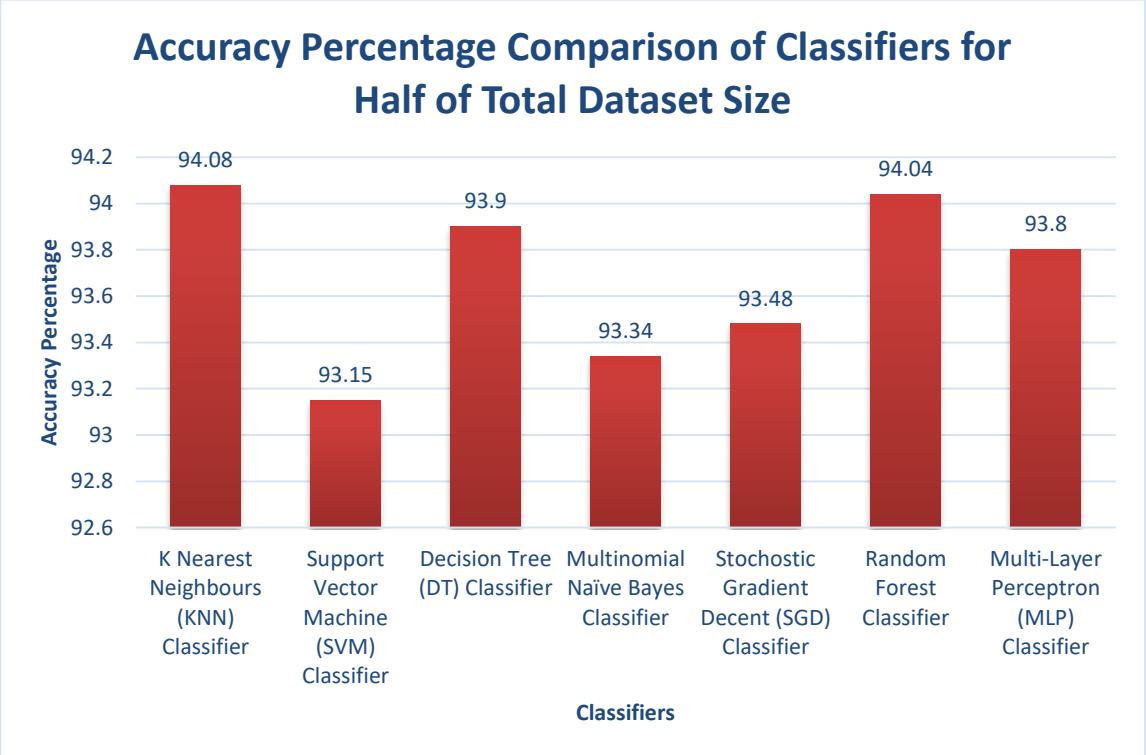


Figure 4.7. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.2 percent) and Half of Total Dataset Size (10712 Images)*

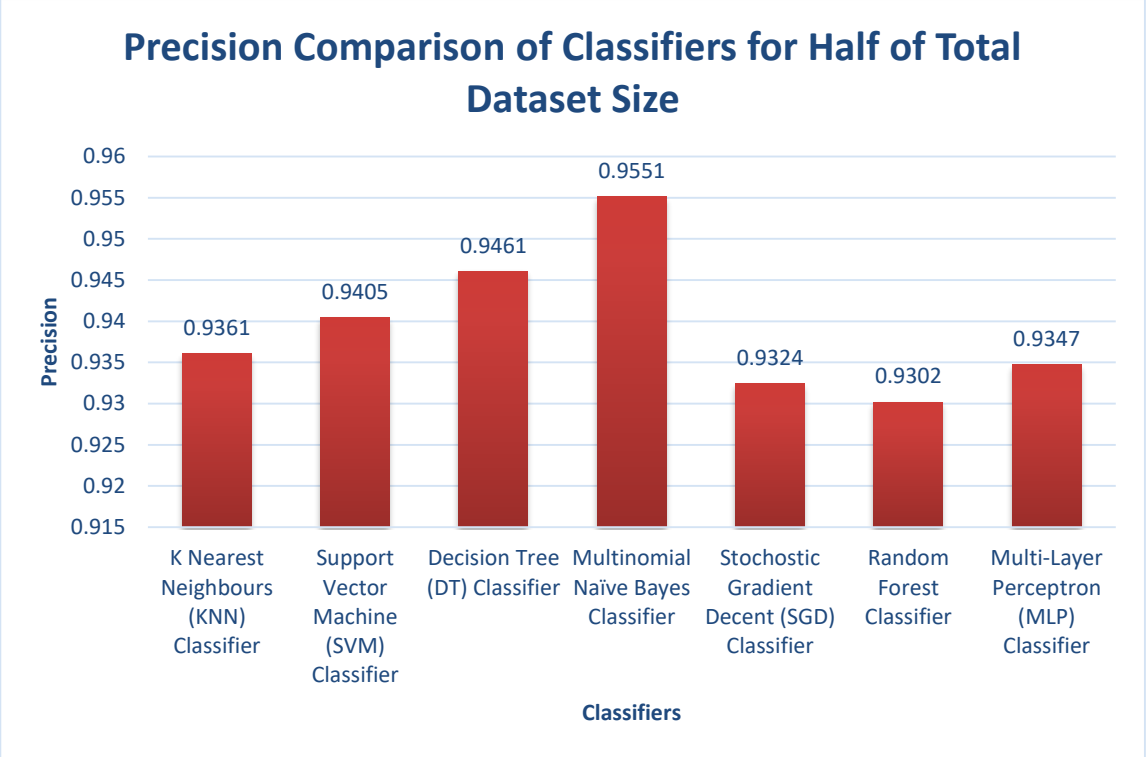


Figure 4.8. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.2 percent) and Half of Total Dataset Size (10712 Images)*

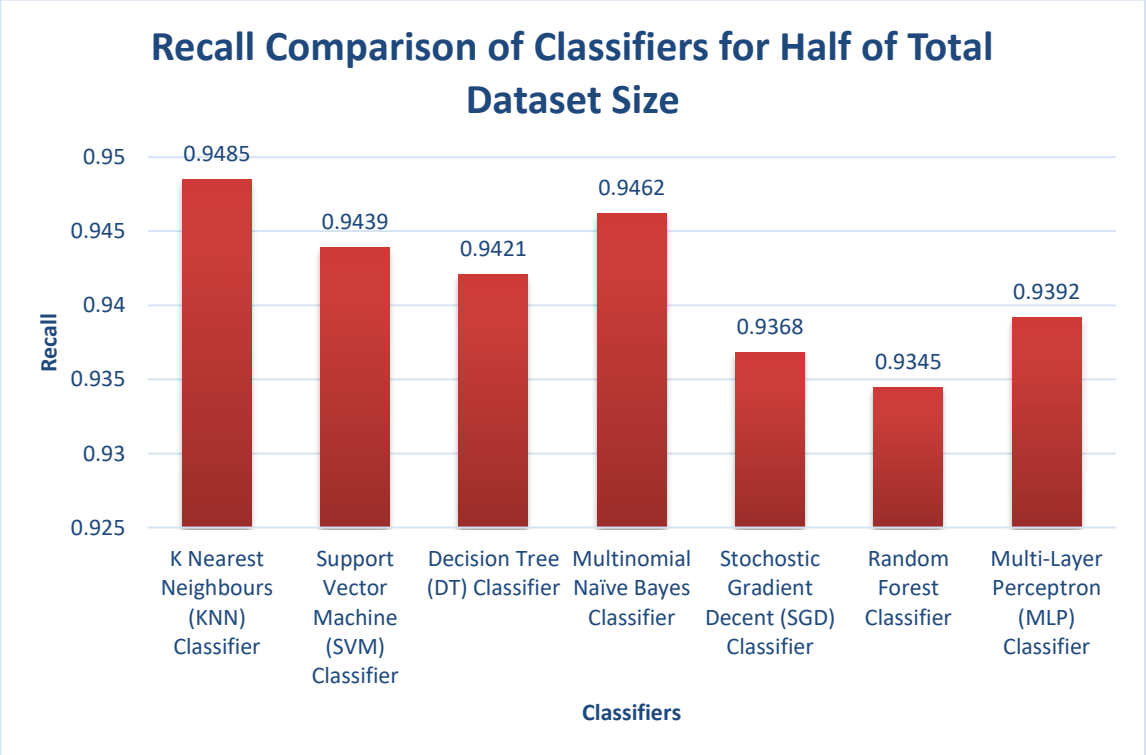


Figure 4.9. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.2 percent) and Half of Total Dataset Size (10712 Images)*

4.2 Constant Test Set Size of 0.3 Percent:

Figures 4.10 to 4.18 show the accuracy, precision, and recall achieved using all the seven classifier when the size of data set was varied according to half of total dataset size, three-fourth of total data set size, and full dataset. The test set size for all three sizes of datasets was fixed at 0.3 percent of the total dataset size used.

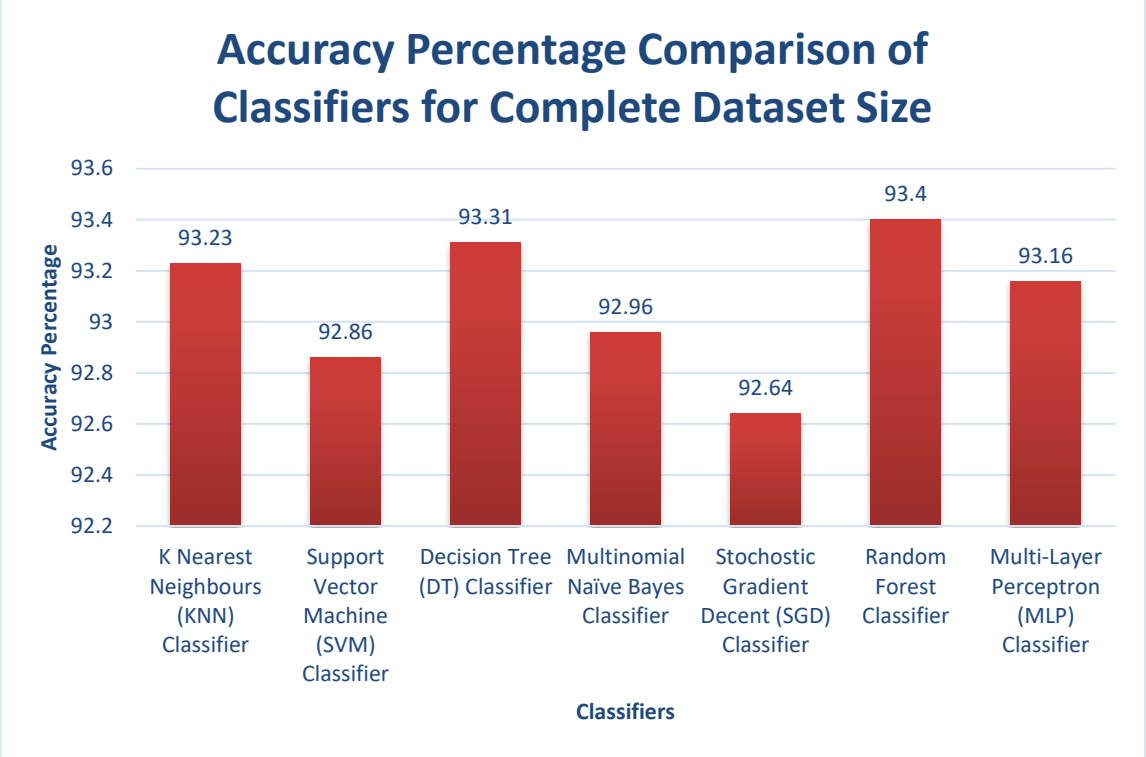


Figure 4.10. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.3 percent) and Complete Dataset Size (21190 Images)*

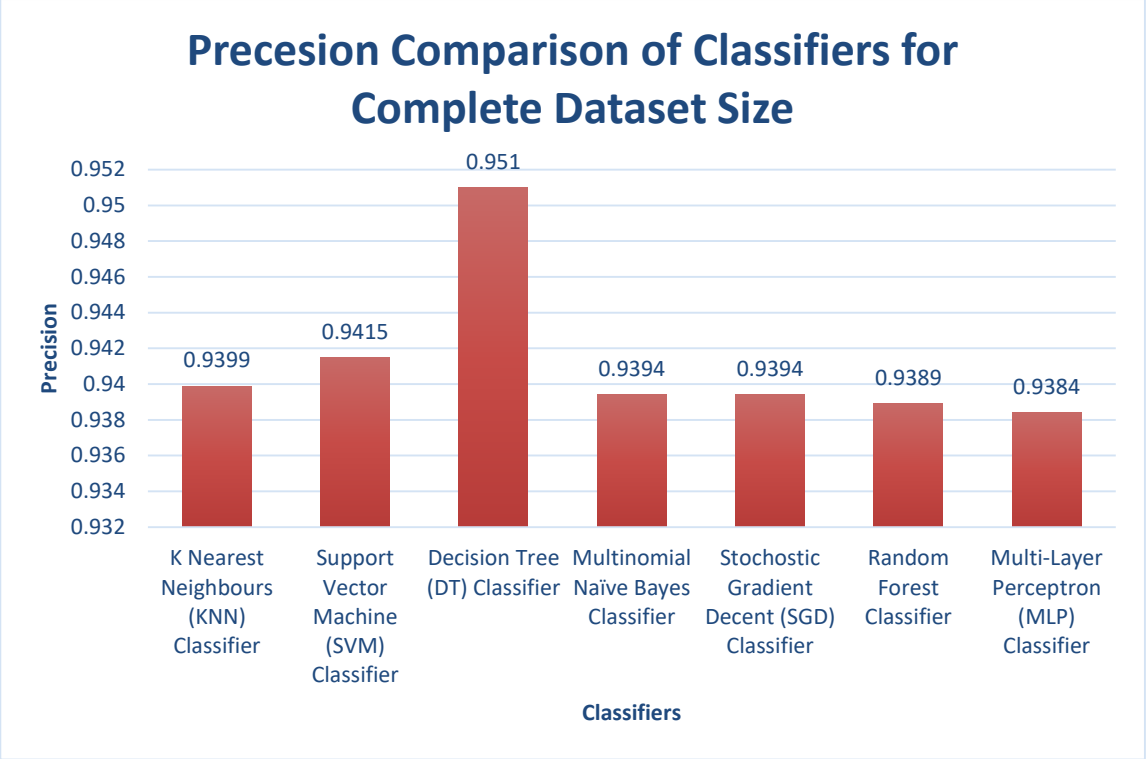


Figure 4.11. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.3 percent) and Complete Dataset Size (21190 Images)*

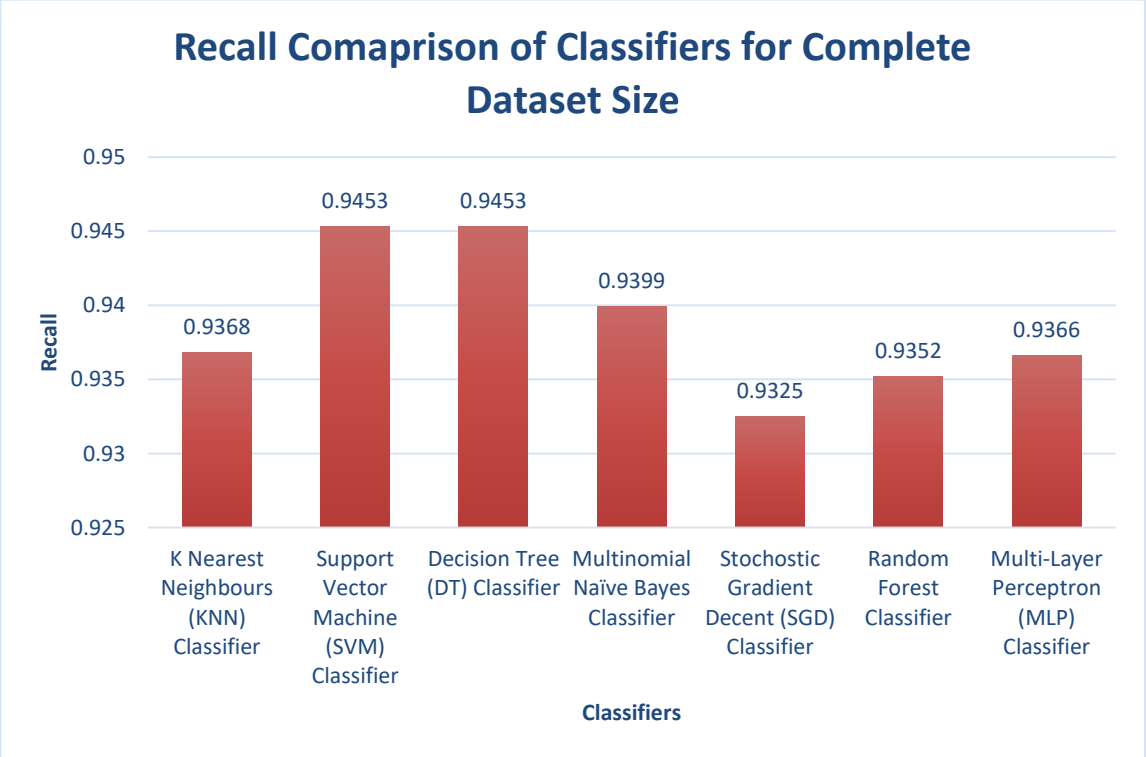


Figure 4.12. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.3 percent) and Complete Dataset Size (21190 Images)*

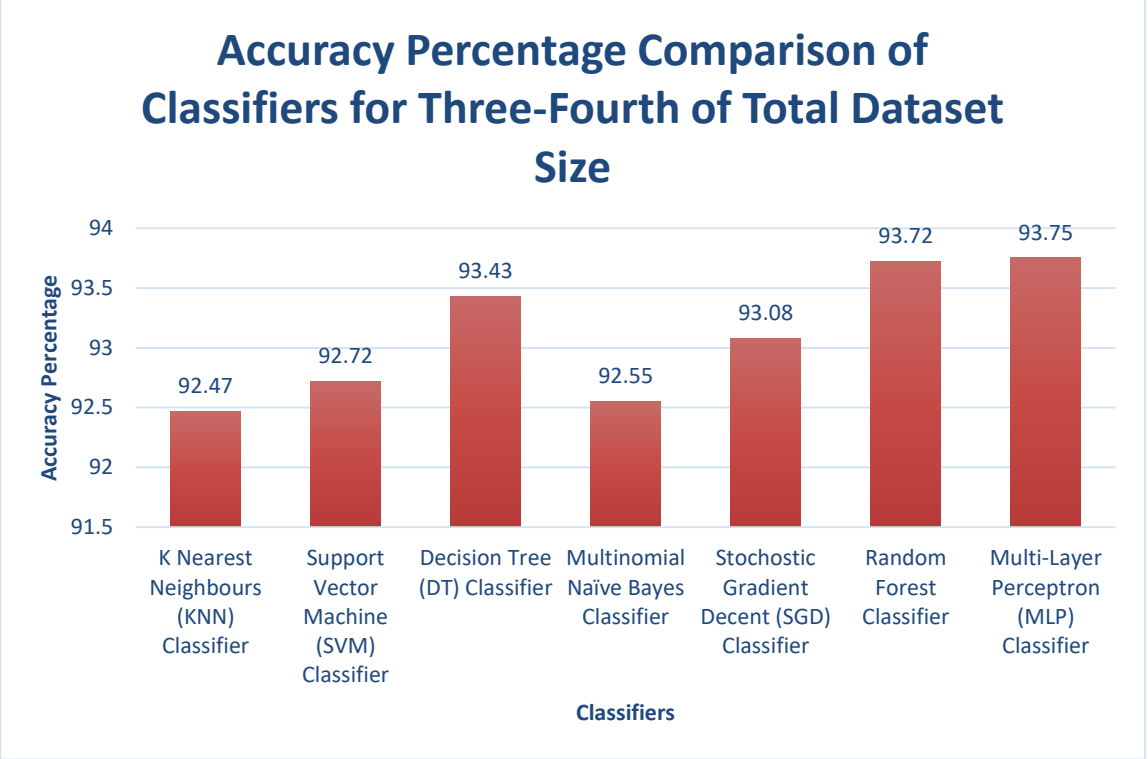


Figure 4.13. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.3 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

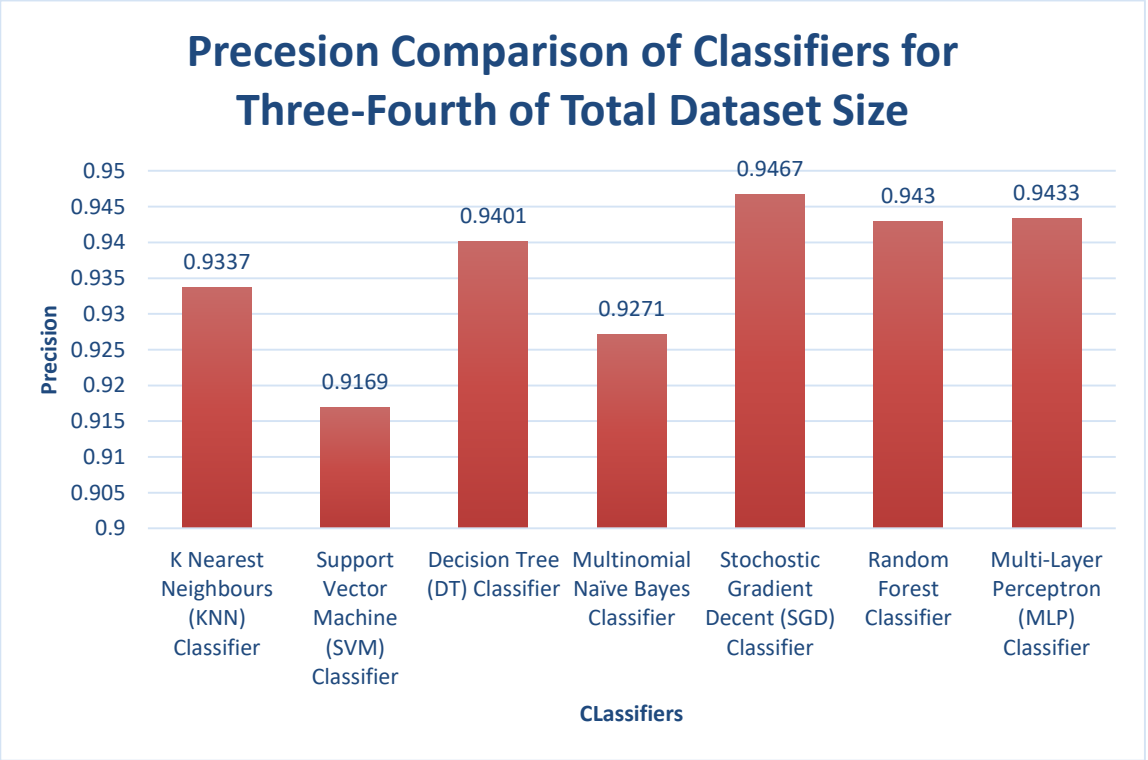


Figure 4.14. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.3 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

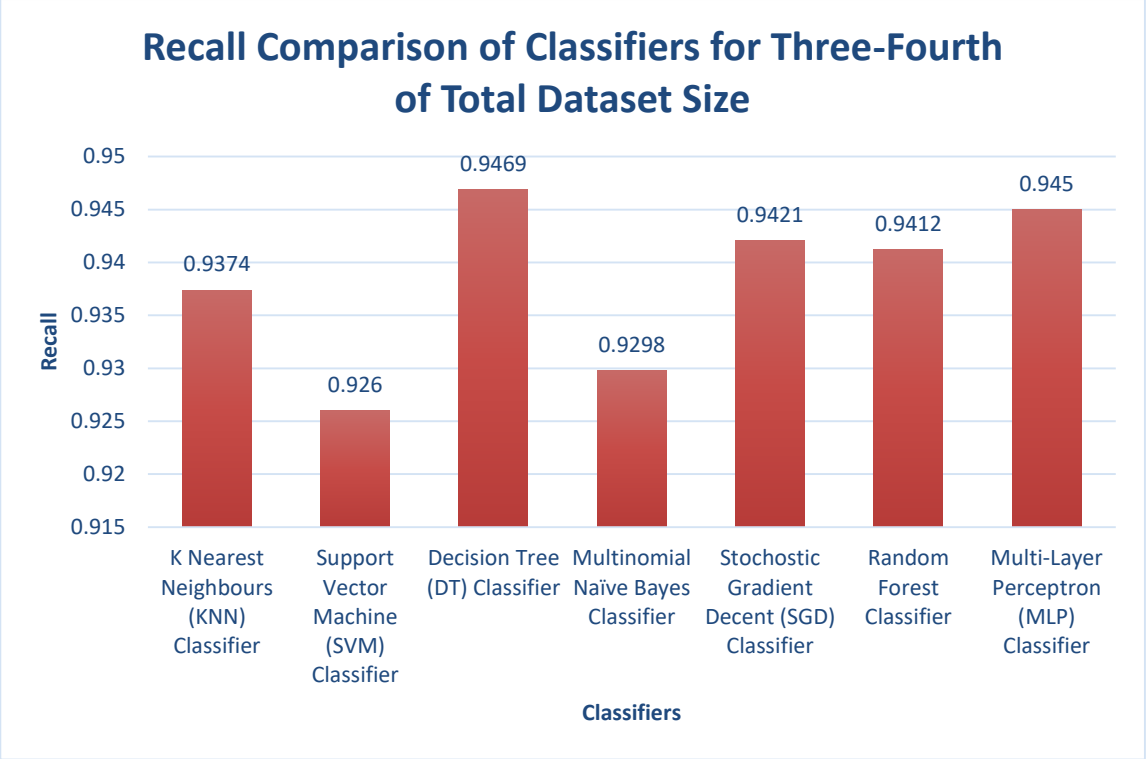


Figure 4.15. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.3 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

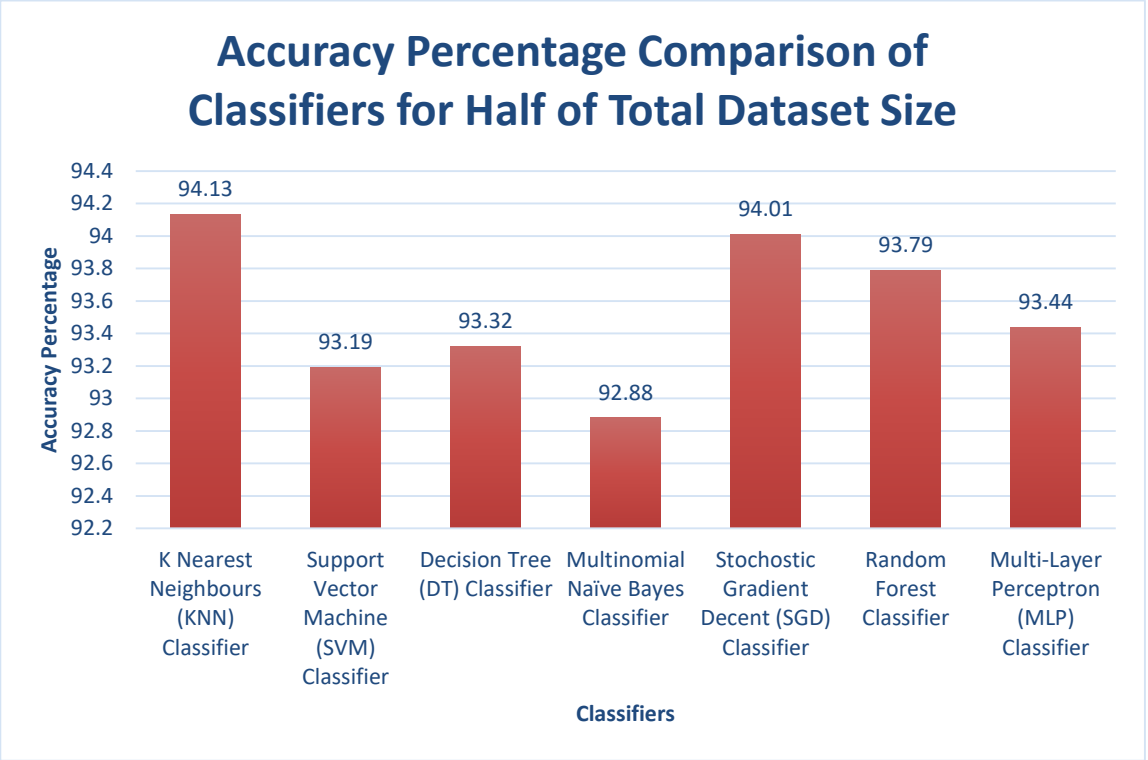


Figure 4.16. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.3 percent) and Half of Total Dataset Size (10712 Images)*

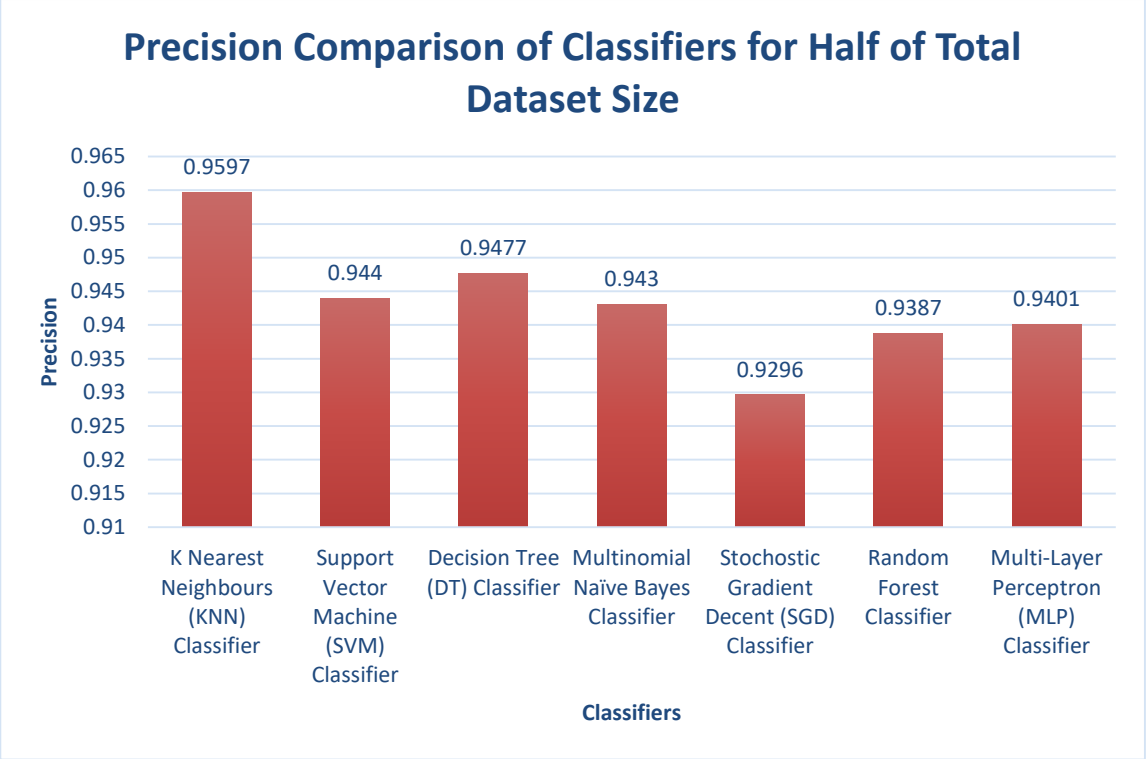
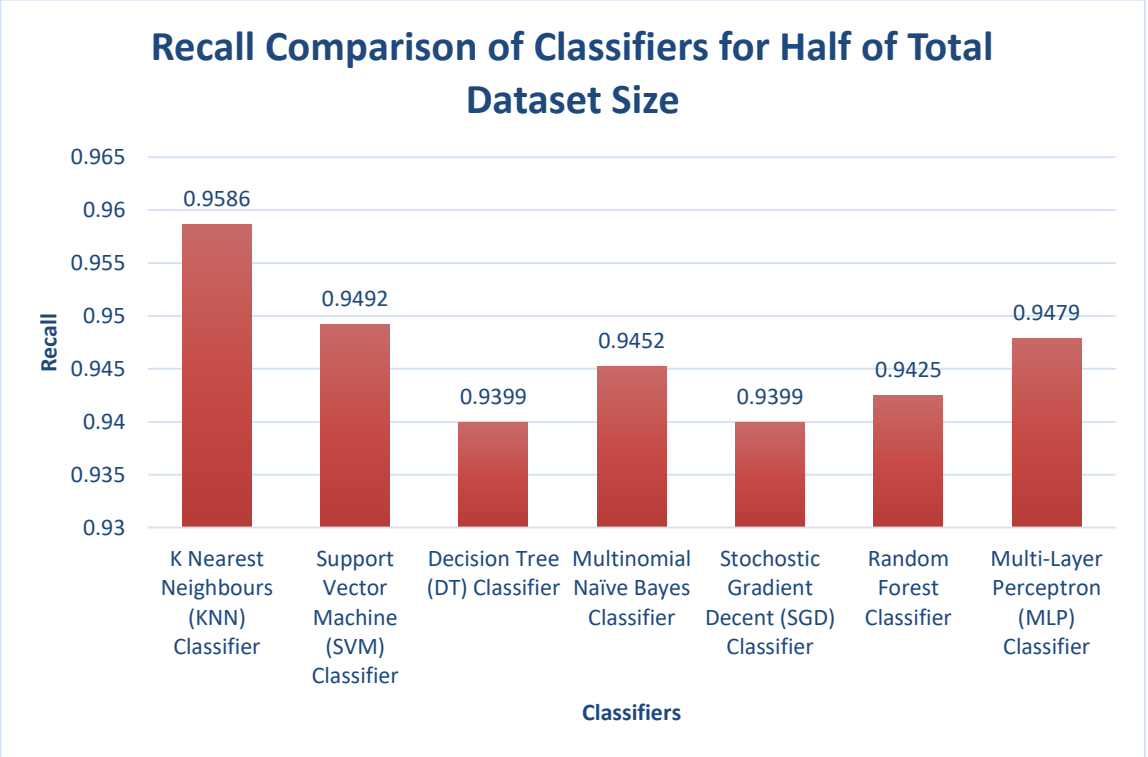


Figure 4.17. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.3 percent) and Half of Total Dataset Size (10712 Images)*



4.18. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.3 percent) and Half of Total Dataset Size (10712 Images)*

4.3 Constant Test Set Size of 0.5 Percent:

Figures 4.19 to 4.27 show the accuracy, precision, and recall achieved using all the seven classifier when the size of data set was varied according to half of total dataset size, three-fourth of total data set size, and full dataset. The test set size for all three sizes of datasets was fixed at 0.5 percent of the total dataset size used.

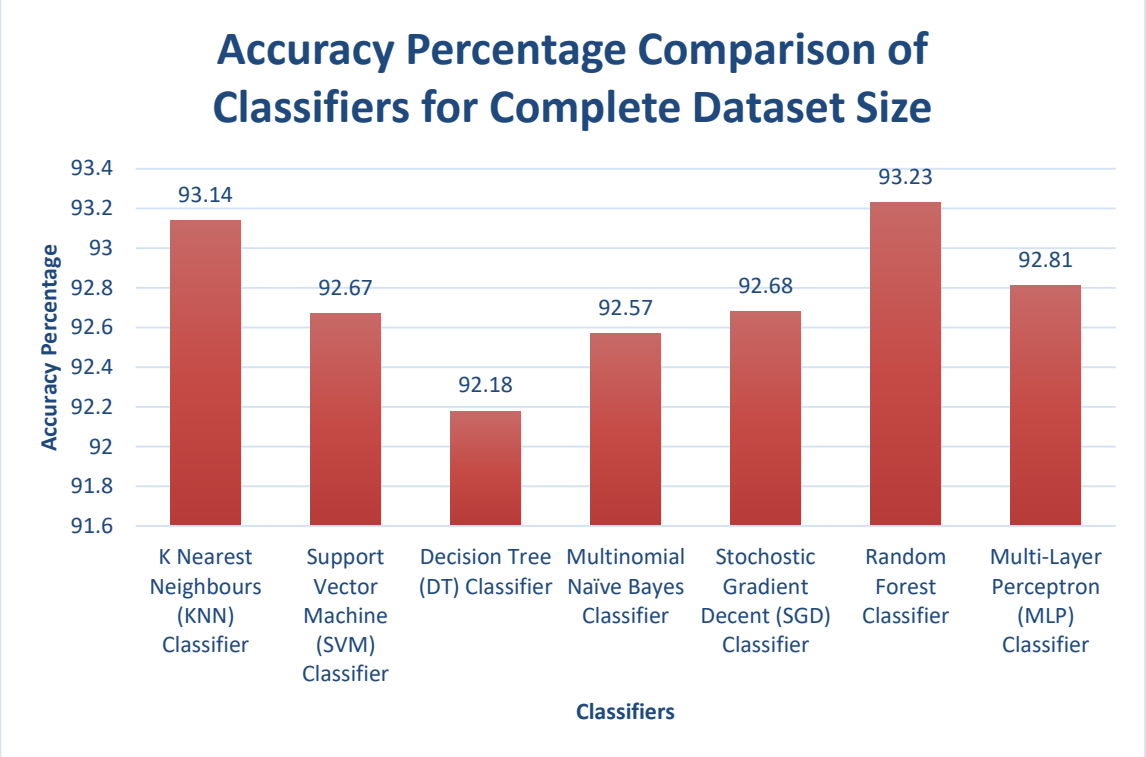


Figure 4.19. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.5 percent) and Complete Dataset Size (21190 Images)*

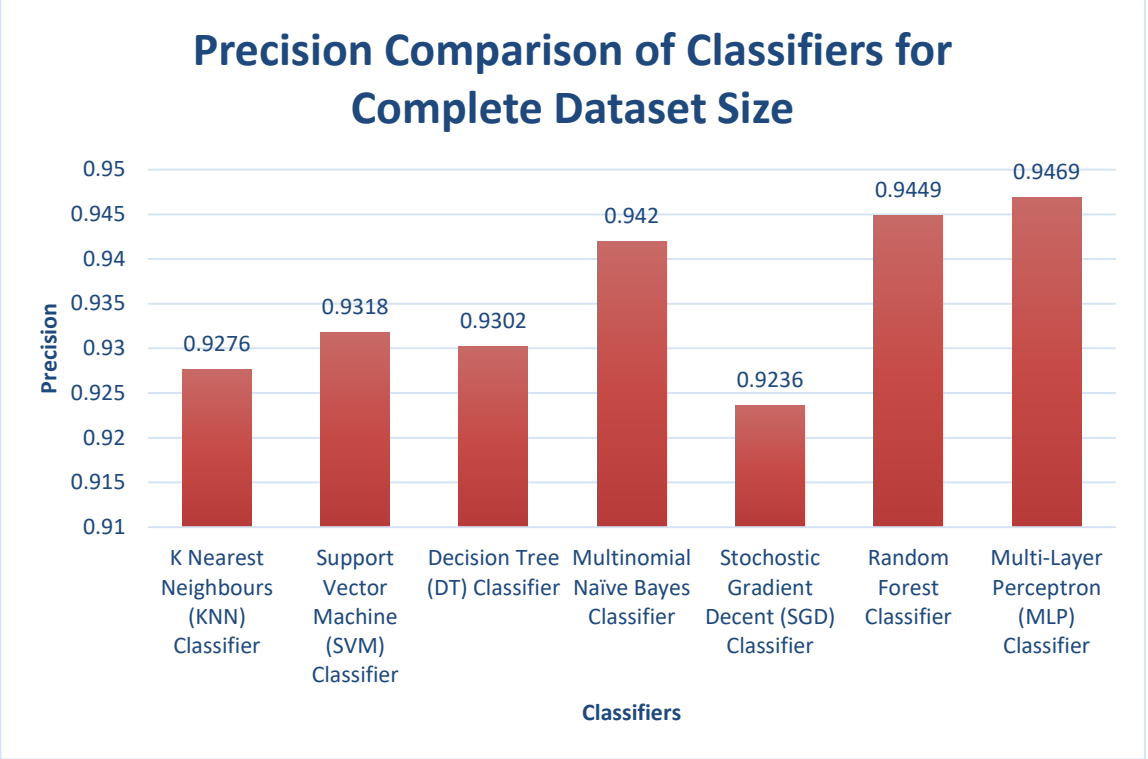


Figure 4.20. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.5 percent) and Complete Dataset Size (21190 Images)*

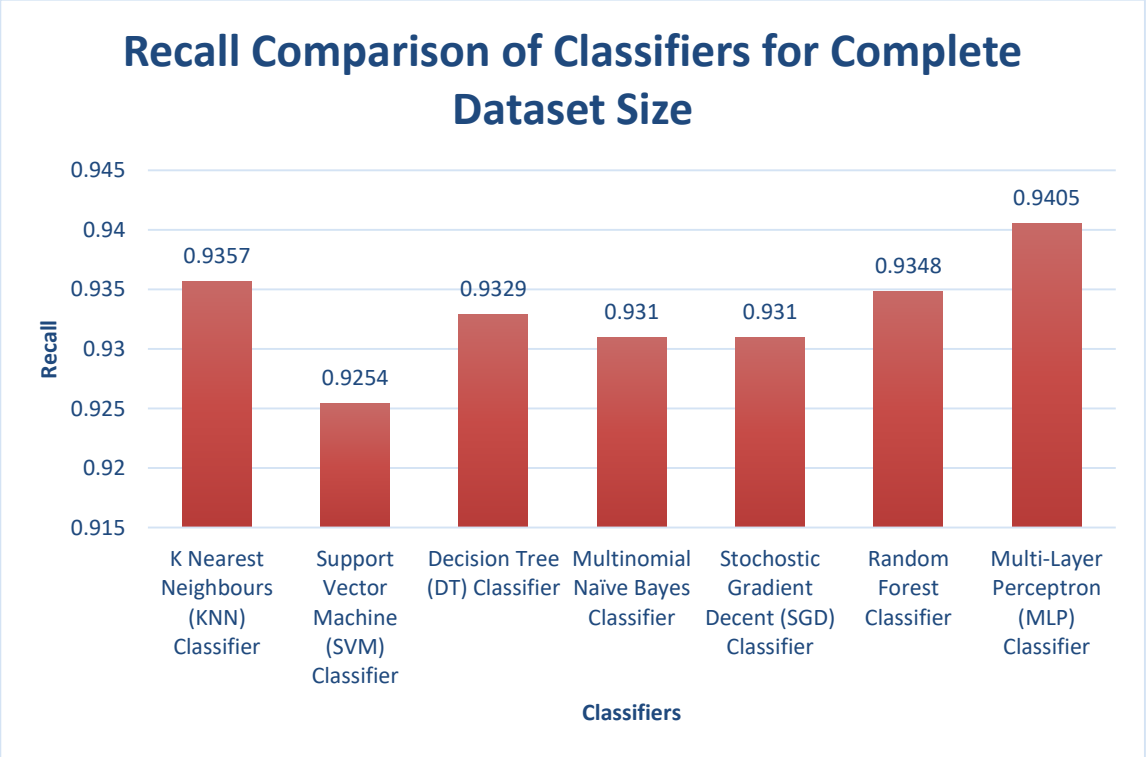


Figure 4.21. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.5 percent) and Complete Dataset Size (21190 Images)*

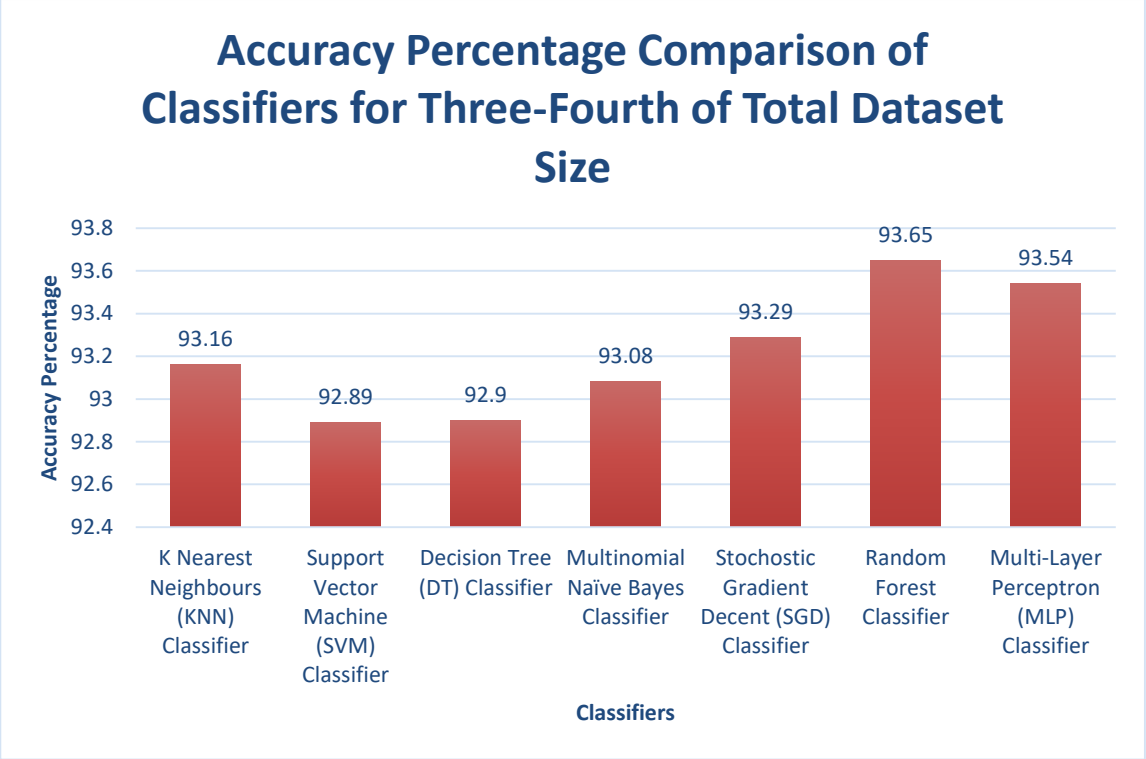


Figure 4.22. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.5 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

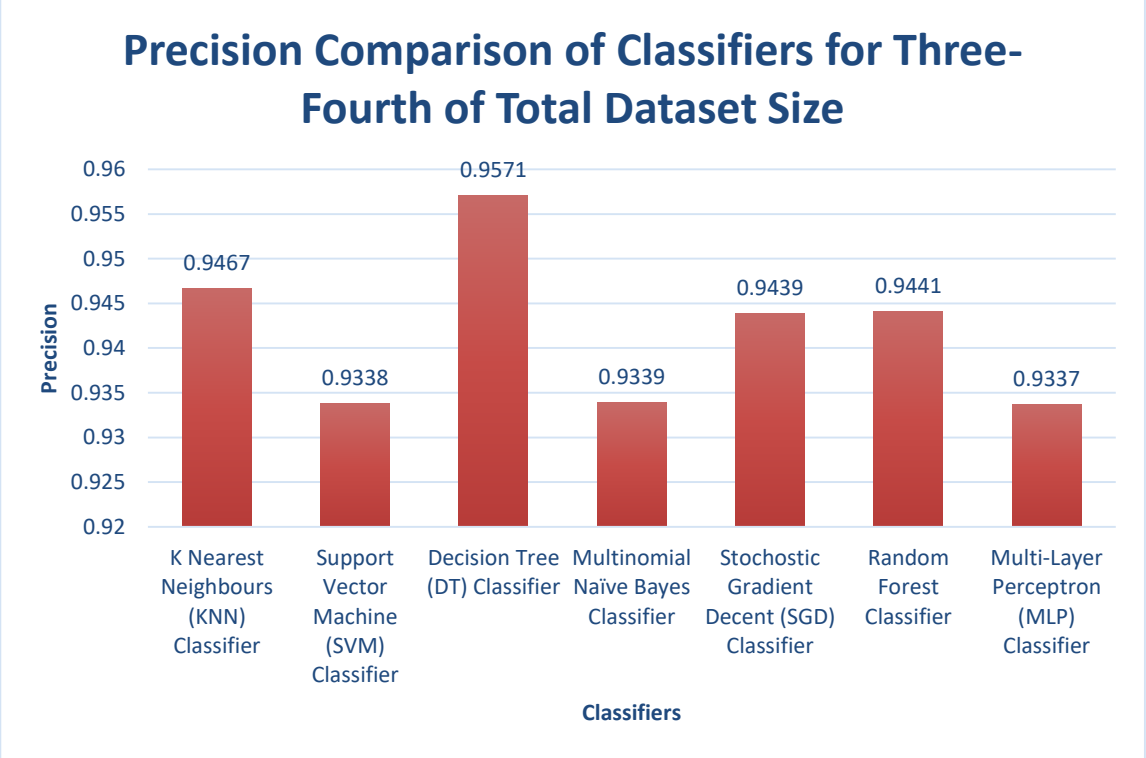


Figure 4.23. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.5 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

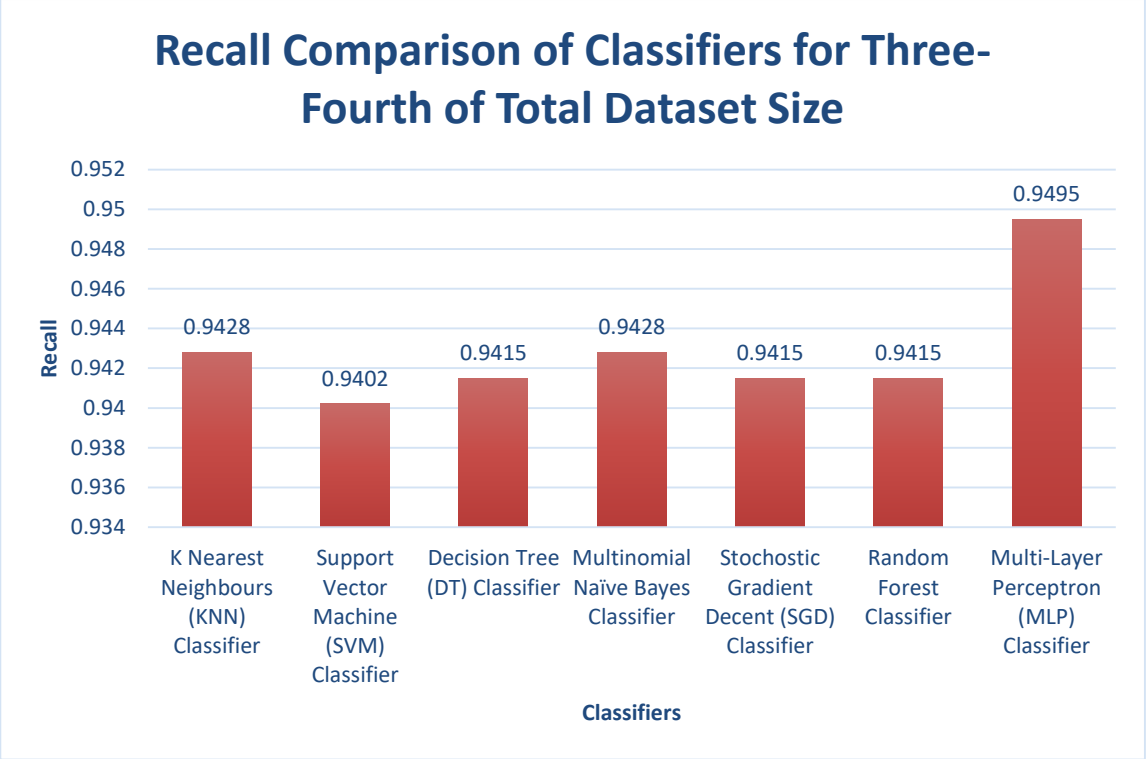


Figure 4.24. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.5 percent) and Three-Fourth of Total Dataset Size (15077 Images)*

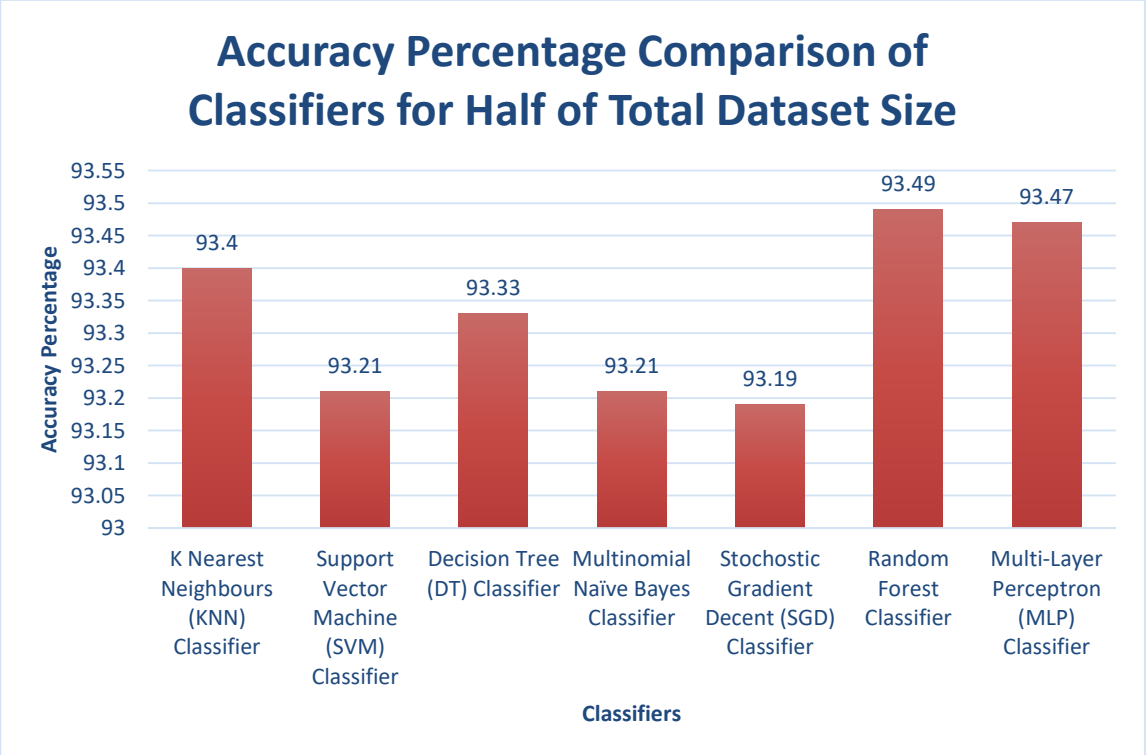


Figure 4.25. *Graph showing Accuracy Percentage Comparison of Various Classifiers with constant test size (0.5 percent) and Half of Total Dataset Size (10712 Images)*

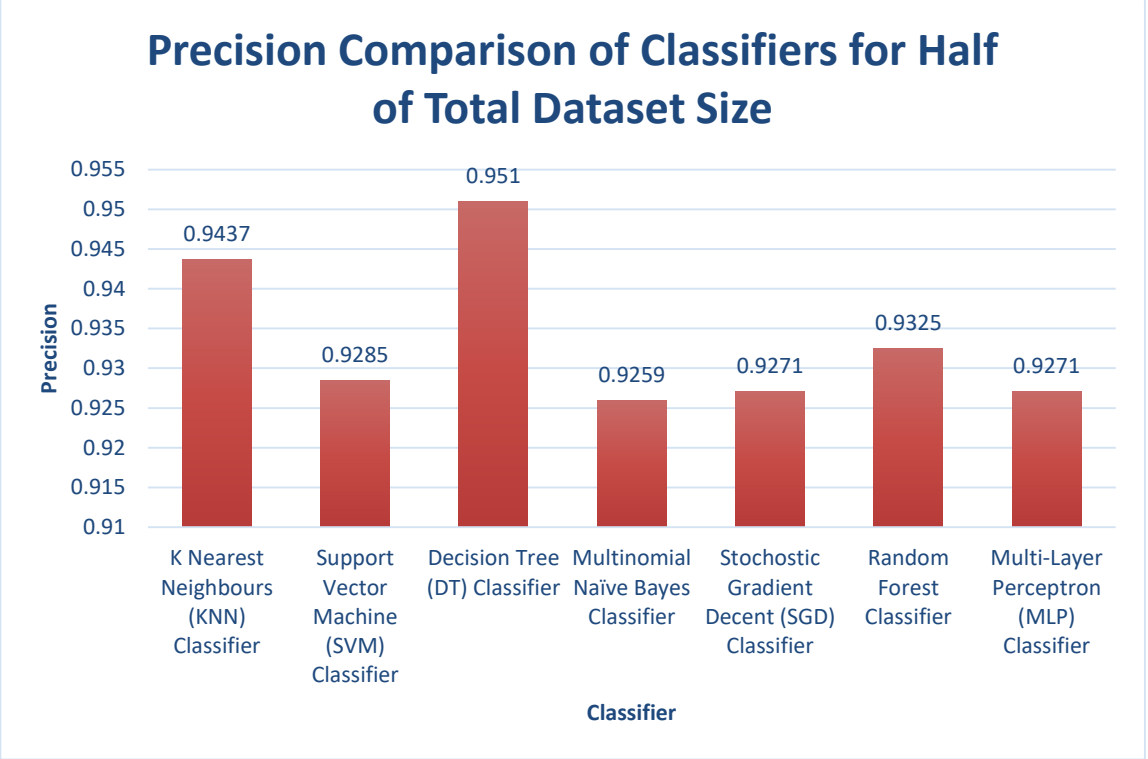


Figure 4.26. *Graph showing Precision Comparison of Various Classifiers with constant test size (0.5 percent) and Half of Total Dataset Size (10712 Images)*

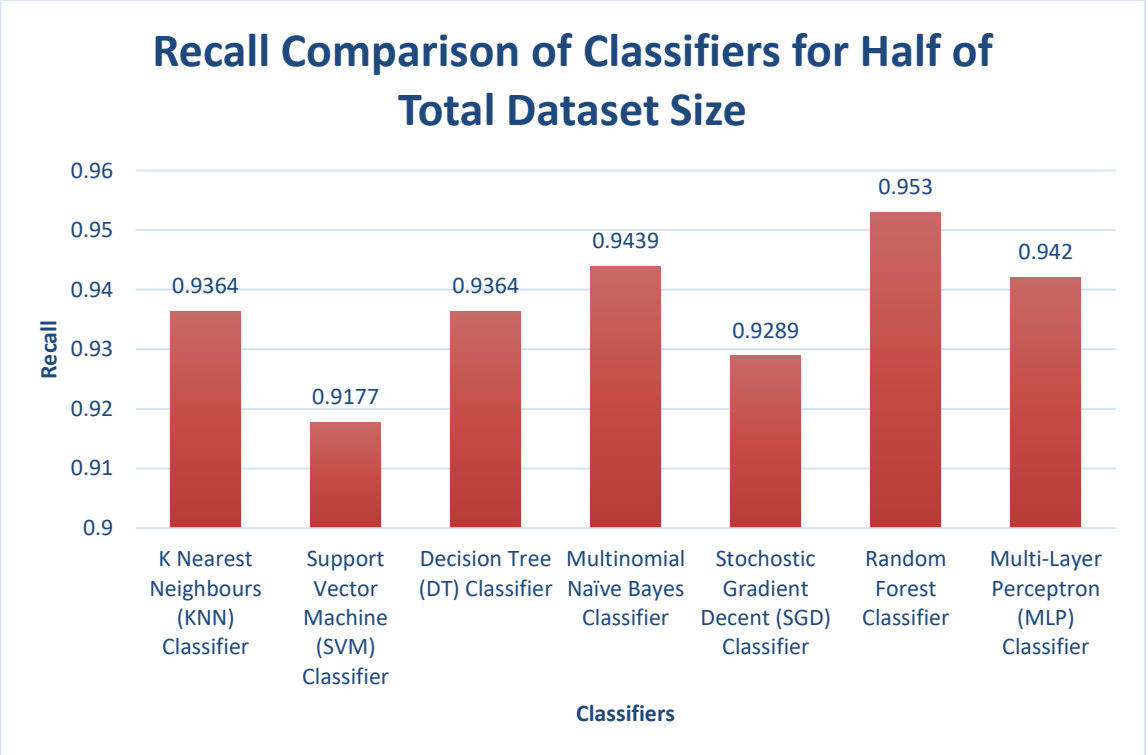


Figure 4.27. *Graph showing Recall Comparison of Various Classifiers with constant test size (0.5 percent) and Half of Total Dataset Size (10712 Images)*

CHAPTER 5: CONCLUSION

A Computer Aided Diagnostic (CAD) system that can use computer vision to provide a second opinion to the radiologist for diagnosis of lung cancer is achievable using Neural Networks and various supervised learning classifiers that will be accurate enough to be introduced in actual practice. But the biggest challenge to reach that objective is to remove the inconsistencies in the inputs that exist due to use different machines from different manufacturers, which are used on patients from different positions and angles; a challenge that does exist when it comes to practical applications of our project. Testing my model with a dataset of size 11.2 GBs indicates good results with a maximum accuracy of 93.73%, precision of 94.38%, and recall of 93.98% obtained by using Random Forest classifier when calculated with complete dataset size and with a test set size of 0.2 percent of the total dataset size.

These image processing techniques can also be used in association with Convolutional Neural Networks (CNN) which is type of unsupervised learning. But in order to that a much larger dataset is required, along with more computational power.

REFERENCES

[1] Indian Cancer Society, Available at: http://www.indiancancersociety.org/ (accessed April, 2017).

[2] WebMd Lung Cancer Health, Available at: http://www.webmd.com/lung-cancer/ (accessed April, 2017).

[3] Computed Tomography (CT) Scan of the Body, Available at: http://www.webmd.com/a-to-z- guides/computed-tomography-ct-scan-of-the-body#1 (accessed April, 2017).

[4] US National Library of Medicine National Institutes of Health, Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1665219/ (accessed April, 2017).

[5] Mokhled S. AL-TARAWNEH, "Lung Cancer Detection Using Image Processing Techniques", Leonardo Electronic Journal of Practices and Technologies, Issue 20, January-June 2012, p. 147-158.

[6] *Non-Small Cell Lung Cancer,* Available at: http://www.katemacintyrefoundation.org/ pdf/non-small-cell.pdf, Adapted from National Cancer Institute (NCI) and Patients Living with Cancer (PLWC), 2007, (accessed July 2011).

[7] Gonzalez R.C., Woods R.E., "*Digital Image Processing"*, Upper Saddle River, NJ Prentice
Hall, 2008.

[8] Venkateshwarlu K., "*Image Enhancement using Fuzzy Inference System"*, in Computer Science & Engineering, Master thesis, 2010.

[9] Jinsa Kuruvilla, K. Gunavathi, "Lung Cancer Classification using Neural Networks for CT Images", International Conference on Computing Science.

[11] Matrices Fritz Albregtsen Image, "Statistical Texture Measures Computed from Gray Level Coocurrence Processing", November 5, 2008.

[12] The flowchart of K nearest neighbor classifier procedure, Available at: https://www.researchgate.net/figure/237080861_fig2_Fig-2-The-flowchart-of-K-nearest-neighbor-classifier-procedure, (accessed April 2017).

[13] Giuseppe Amato, Fabrizio Falchi, "kNN based image classification relying on local feature similarity", Third International Workshop on Similarity Search and Applications, SISAP 2010, September 2010, p. 18-19.

[14] Laurence Smith, John Tansley, “DECISION TREE ANALYSIS”, United States Patent Application Publication, Oct. 7, 2004.

[15] Neha Patel, Divakar Singh, “An Algorithm to Construct Decision Tree for Machine Learning based on Similarity Factor”, International Journal of Computer Applications, Volume 111 – No 10, February 2015, p.22.

[16] Joachims, Thorsten, “Making large-scale SVM learning practical”, Technical Report, Universität Dortmund, 1998, p.28.

[17] Shuo Xu, “Bayesian Naïve Bayes classifiers to text classification”, Journal of Information Science, 2016, p.1–12.

[18] Leo Breiman, “RANDOM FORESTS”, Springer Netherlands, volume 45, January 2001, p. 5-32.

[19] Morphological Transformations, Available at: http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html, (accessed December 2016).

[20] Documentation – graycoprops, Available at: http://in.mathworks.com/help/images/ref/graycoprops.html, (accessed April 2017).

[21] A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm, Available at: https://saravananthirumuruganathan.wordpress.com/2010/05/17/a-detailed-introduction-to-k-nearest-neighbor-knn-algorithm/, (accessed April 2017).

[22] Introduction to k-nearest neighbors: Simplified, Available at: https://www.analyticsvidhya.com/blog/2014/10/introduction-k-neighbours-algorithm-clustering/, (accessed April 2017).

[23] Euclidean distance - n dimensions, Available at: https://en.wikipedia.org/wiki/Euclidean_distance#n_dimensions, (accessed April 2017).

[24] sklearn.neighbors.KNeighborsClassifier , Available at: http://scikitlearn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html, (accessed April 2017).

[25] Decision Tree – Classification, Available at: http://www.saedsayad.com/decision_tree.htm, (accessed April 2017).

[26] 6 Easy Steps to Learn Naive Bayes Algorithm (with code in Python), Available at: https://www.analyticsvidhya.com/blog/2015/09/naive-bayes-explained/, (accessed April 2017).

[27] Stochastic Gradient Descent, Available at: http://scikit-learn.org/stable/modules/sgd.html, (accessed April 2017).

[28] sklearn.linear_model.SGDClassifier, Available at: http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html#sklearn.linear_model.SGDClassifier, (accessed April 2017).

[29] Random Forests - Leo Breiman and Adele Cutler, Available at: https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm, (accessed April 2017).

[30] 3.2.4.3.1.sklearn.ensemble.RandomForestClassifier, Available at: http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html, (accessed April 2017).

[31] Neural network models (supervised) - Multi-layer Perceptron, Available at: http://scikit-learn.org/stable/modules/neural_networks_supervised.html, (accessed April 2017).

[32] sklearn.neural_network.MLPClassifier, Available at: http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html, (accessed April 2017).