SEMI AUTOMATED SYSTEM FOR PLANT LEAF DISEASE DETECTION BASED ON SEGMENTATION, FEATURE EXTRACTION AND ENSEMBLE CLASSIFIER

A Thesis

Submitted in partial fulfillment of the requirements for the

award of the degree of

DOCTOR OF PHILOSOPHY

In

Computer Science Engineering

By

Navneet Kaur

41700044

Supervised By

Dr. V. Devendran

LOVELY PROFESSIONAL UNIVERSITY

PUNJAB

2022

Declaration

I hereby affirm that the thesis entitled "Semi automated system for plant leaf disease detection based on segmentation, feature extraction and ensemble classifier" submitted by me for the Degree of Doctor of Philosophy in Computer Science and Engineering is the result of my original and independent research work carried out under the guidance of Supervisor Dr. V. Devendran and it has not been submitted for to any university or institute for the award of any degree or diploma.

Place:

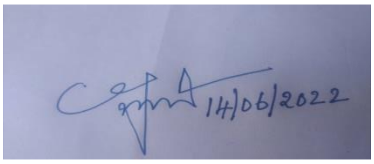
Date:

Signature of the Candidate

Certificate

This is to certify that the thesis entitled "Semi automated system for plant leaf disease detection based on segmentation, feature extraction and ensemble classifier" submitted by Ms. Navneet Kaur for the award of the degree of Doctor of Philosophy in School of Computer Science and Engineering, Lovely Professional University is entirely based on the work carried out by her under my supervision and guidance. The work reported embodies the original work of the candidate and has not been submitted to any other university or institution for the award of any degree or diploma, according to the best of my knowledge.

Date: 14/06/2022



Signature of the Supervisor

ABSTRACT

In recent decades, detecting plant leaf diseases using machine learning algorithms has constantly improved. Plant health is an important aspect of sustainable agriculture. Leaves are one of the most sensitive components of a plant and they are usually the first to succumb to the infection. A mechanism for disease diagnosis with the least amount of human interaction has been proposed. All of this is done in a matter of milliseconds via image recognition. This research uses multiple image processing stages, such as segmenting an image, extracting features, and classifying the image to diagnose diseases in plant leaf diseases. Numerous state-of-the-art methodologies have been examined and hence a semi-automated system is proposed that takes the user's input of the leaf image and predicts the disease. The user is required to select the plant type from the given plant options and upload the image. The uploaded image passes through the testing phase that involved segmentation, feature extraction and classification stages. The appropriate disease is then predicted. Less focus has been made upon hybrid approaches in existing approaches and no convolutional based features have been completely ignored in the state-of-the-art techniques. We have stressed on hybridization of different feature extraction algorithms and machine learning classifiers for the accurate prediction of diseases. Using the PlantVillage dataset, in this research, a very novel approach is proposed that focuses on Law's texture features, which actually involves a convolution process for extracting features and is an acknowledged feature extraction algorithm in machine learning. Along with this, other techniques as Grey Level Co-occurrence Matrix, Local Binary Pattern and Gabor features have been conjunct to create a hybrid approach for robust results. For segmentation of diseased regions, K-means segmentation algorithm is utilized which has been optimized using Grey Wolf Optimization. For classification, we have utilized ensemble learning. The best ensemble is investigated to obtain the best results. A total of three crop types have been considered for the study – Bell Pepper, Potato, and Tomato. Two categories from bell pepper, three from potato, and ten categories from tomato are considered in this work. We have compared our approach to existing approaches. Our approach has proved to be robust as compared to various existing approaches and has revealed an average accuracy of 92.67%

ACKNOWLEDGEMENTS

Dr. V. Devendran, my research advisor, has been a tremendous source of inspiration during this research work. He gave me a lot of leeway in choosing a research topic that I was interested in, and he directed and constructively criticised me when I needed it. The completion of this dissertation would not have been feasible without his inspiration, tireless efforts, passionate attitude, and personal attention. His insightful input, rational reasoning, and suggestions were quite beneficial to me. His involvement was crucial in the creation of this piece. He has been a huge source of strength for me, and I want to express my gratitude to him from the bottom of my heart.

I would like to express my gratitude to the Lovely Professional University (Punjab), India's School of Computer Science and Engineering for providing this doctoral programme. Throughout my research period, the institute and division were always very helpful in providing all of the necessary services and equipment. I am also thankful to Honourable Chancellor Mr. Ashok Mittal, Worthy Pro Chancellor Mrs. Rashmi Mittal, Executive Dean, Faculty of Technology & Sciences Dr. Loviraj Gupa, Head of the Department Mr. Pushpendra Kumar Pateriya, my colleagues and friends for supporting me.

I am grateful to all of the reviewers who provided helpful feedback and comments on my research publications, which helped me to improve the quality of this research. I want to express my gratitude to all writers referenced in my thesis, whose study findings offer me with a battleground from which to choose study directions.

I appreciate my parents' well-wishes and words of support. They have encouraged me on my arduous path to obtaining a doctorate degree.

Navneet Kaur

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
ANFIS	Adaptive Neuro Fuzzy Inference System
ANN	Artificial Neural Network
AUC	Area Under the Curve
BBP	Bin Binary Pattern
BPNN	Back Propagation Neural Network
BS	Blue Squares
BoVW	Bag of Visual Words
CBIR	Content-Based Image Retrieval
CCF	Comprehensive Color Feature
CIE	International Commission on Illumination
CNN	Convolutional Neural Network
DCNN	Deep convolution neural network
DCT	Discrete Cosine Transform
DE	Differential Evolution
DFT	Discrete Fourier Transforms
DICNN	Dense Inception Convolutional Neural network
DLP	Disease-Level-Parameter
DSI	Disease-Severity-Index
DSIFT	Dense Scale Invariant Feature Transform

DT	Decision Tree
DTW	Deterministic Tourist Walk
DWT	Discrete Wavelet Transform
EM	Expectation Maximization
ExG	Excess Green
FCM	Fuzzy C-Means
FK	Fisher Kernel
FN	False Negative
FRVM	Fuzzy Relevance Vector Machine
FP	False Positive
FV	Fisher Vector
GA	Genetic Algorithm
GB	Green minus Blue
GC	Green Circles
GCH	Global Color Histogram
GDP	Gross Domestic Product
GLCM	Gray-Level Co-occurrence Matrix
GLM	Generalized Linear Model
GSA	Gravitational Search Algorithm
GWO	Grey Wolf Optimization
HSV	Hue Saturation Value
HIC	Histogram Information Content

HIS	Hue Intensity Saturation
HOG	Histogram of Oriented Gradient
IOT	Internet of Things
IPR	Infection-Per Region
LDA	Linear Discriminant Analysis
KNN	K nearest neighbor
LBP	Local Binary Pattern
LCH	Local Color Histograms
MD	Mahalanobis Distance
MDcE	Manhattan Distance controlled Entropy
MLP	Multilayer Perceptron
MSE	Mean-squared Error
M-SVM	Multi-Class Support Vector Machine
NB	Naïve Bayes
NN	Neural Network
PCA	Principal Component Analysis
PDbE	Probability Distribution-based Entropy
PHOG	Pyramid Histogram of Gradients
PHOW	Pyramid Histograms Of Visual Words
PNN	Probabilistic Neural Network
PSNR	Peak Signal-to-Noise Ratio
PSO	Particle Swarm Optimization

RBF	Radial Basis Function
ReLU	Rectified Linear Units
RF	Random Forest
RGB	Red Green Blue
ROC	Receiver Operating Characteristic
ROI	Region of Interest
RS	Red Star
SIFT	Scale-Invariant Feature Transform
SMOTE	Synthetic Minority Over-sampling Technique
SOS	Symbiotic Organism Search
SPAM	Subtractive Pixel Adjacency Model
SURF	Speeded-Up Robust Features
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TCCNN	Three Channel Convolution Neural Network
TN	True Negative
TP	True Positive
WSN	Wireless Sensor Network

CHAPTER 1
INTRODUCTION

Image processing is a technique for applying operations on an image as an attempt to extract relevant information from it. It's a process in which the input is an image and the output is either that image or its characteristics/features. It has found a number of applications such as remote sensing, medical imaging, agriculture, face detection, finger print detection and forecasting. In agriculture, image processing can be used for bug detection, nutrient shortages and plant content identification, fruit quality control, rating of farm products such as fruits and vegetables, crop and land assessment, and object recognition, among other things. This chapter discusses the background study, phases in leaf disease detection, motivation, problem statement, research objectives and thesis contribution. In the last section the complete thesis organization is also described.

1.1 BACKGROUND STUDY

Agriculture is considered as one of the chief sources of livelihood for people around the globe. And India is undoubtedly an agricultural generator globally. Agriculture in India started from the Indus Valley Civilization [111]. India has been categorized second globally in the agronomic products. According to [54], more that 50 percent of manpower was deployed in agriculture and committed to 17-18% of the GDP of India. Farming is a primary source of income for about 58% of the population of India. India stands as the sixth largest in food and packaged goods and contributes to the 70% of the sales.

Artificial Intelligence [81] has helped for agriculture revolution. Considering agriculture as a significant origin for economic growth [1], heterogeneous plants are reaped as per the requirement of the country and also the abode circumstances and conditions. But, there can be several issues that the farmers around the globe face and that may comprise water shortage, bad weather conditions, natural disasters and plant diseases. Out of all these existing problems, problem of plant diseases detection can be resolved with the help of technical aids in the form of machine learning in general and image processing in specific. As it may not be possible to remember the information related to each and every type of disease in plants since there are very few specialists involved in this area, so this type of technical aid will act as a boon for the agriculturists and farmers around the globe.

In most of the developing countries in the world, the vertebra of economy [2] is supported by agriculture. The plant sprouting determines the quality and quantity of a crop. Diseases in crops, most often appear on the most delicate parts, that is the leaf. Intense care is required in order to reveal the diseases as timely as possible. A well timed detection can inhibit the proliferation of diseases to the whole field or bunch of crops which in turn will improve yield [3]. Observing the color and surface of the leaves is the traditional practice to detect the disease. Thereupon, due to time-consuming nature and requirement of regular endeavor and knowledge, the practice fails when it comes to disease detection in huge fields.

A specific plant can show contrasting symptoms for different diseases. It may be the color, texture or shape [25] of the diseased area ranging from small-sized spots to large-sized patches. Due to varying characteristics of a disease, there is necessity for adopting an approach that is less time consuming and therefore boosts up the crop production rate and the economy as well. Disease can be detected on any part of a plant, be it leaves, stem or the fruit itself, a timely revelation is definitely required. A large bunch of researchers have been devoting their time in this domain in order to help out farmers.

Dependency on the production of crop [8] for economic growth cannot be ignored and hence constitutes an important part of it. For the purpose of boosting the rate of production, the most important thing to take into consideration is monitoring the diseases related to different crops within intended deadlines. Observation of plant well-being based on the obvious or noticeable indications of disease on the leaves of plants is an indispensable part of agriculture as plants play a dominant role in our ecosystem, imperishable agriculture and atmospheric conditions. Since at the global scale, they are a dominant source of food and economy, it is both a societal and a phenomenal engagement. Significant operations in plants such as photosynthesis can be altered by a disease which can not only affect the quantity of agricultural production but also the quality of the crop. This can further affect the economy of a country like India where agronomy is the dominant source of income for farmers working at small scales. The disease symptoms in a plant generally appear on the leaves. It is very challenging task when it comes to scrutinizing the syndromes of leaves infected by diseases with the help of naked eye [48]. The pathogens and weed are the main reasons of cause of these infections and can't be detected with the traditional optical method. However, it absolutely depends on the experience of the people involved in this field which otherwise, can be time consuming and cumbersome task.

In agronomic productivity, a well-timed prognosis of the disease is the key requirement. If this timely prognosis of the disease is not considered seriously, then crop yield loss is an obvious adverse effect. This is the reason farmers around the world face problems to identify disease symptoms in the diseased plants in a proper manner in the initial stage of the disease.

In order to support these people in identifying the correct diseases in plant leaves, advanced techniques comprising machine learning [7] play a significant role. The recognition system based on machine learning will be a time saver and also will be more efficient as compared to the human assessment system. Therefore, research motivated us to work on bringing up a system that can identify leaf infections in a timely manner in order to avoid any kind of loss in the agronomic productivity and increase it in the best probable way. Hence, the growers can take judicious steps to treat the infections involved in leaves of the plants. Production maximization is the key aim of plant leaf disease detection system.

1.2 PHASES OF PLANT LEAF DISEASE DETECTION

The following figure (Fig 1.1) depicts the various phases involved in plant leaf disease recognition.

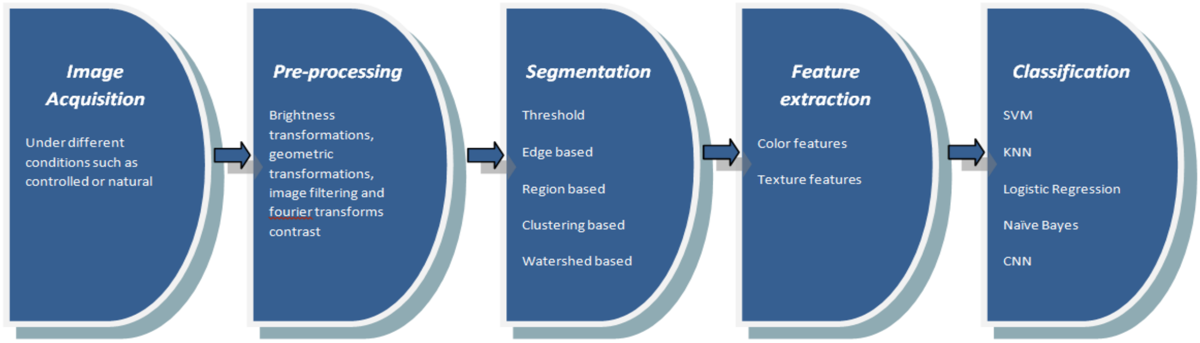


Fig 1.1 Phases of plant leaf disease detection

1.2.1 Image Acquisition

From the name itself, it is clear that an image must be successfully captured. It can be categorized as a phenomenon of retrieval of image, mainly with the aid of hardware deployed source. Later on it can be passed through required procedure for image processing. It is the very first stage in the sequence of progress, since without the image gathering no further processing is practicable. Whatever the image is retrieved is absolutely raw, means it is an unprocessed image and it is dependent on the device that is used in order to acquire that image for creating the baseline for further processing. This stage's one of the final aim is to have a device or a source for input that can operate within the controlled environment such that the image can be nearly reproduced perfectly enough under the same situation or condition. Considering the domain of work to be accomplished, a super factor in the image acquisition is most of the time is the inceptive setup and a constant maintenance of the hardware that is used to photograph the images. The hardware devices can range from simple scanners to high end telescopes. If the image capturing device is not configured appropriately or as required, then the visual traces can be produced that can further make the image processing task more complex. Ill designed setup or not properly configured setup can produce images having low quality such that they cannot be enhanced even with substantial processing. One type of image acquisition in image processing is referred to as real time image acquisition. It refers to a source or a device which is required to capture images automatically without much human intervention.

1.2.2 Pre-processing

This stage is also referred to the process of data cleansing and is a crucial phase in image processing. A dedicated part of the time is required to be spent on this stage. It is also a term used for images at the lowest degree of abstraction. The goal is to enhance the image data in such a way that lousy distortions are suppressed or alternatively it must improve those features that are required for further analysis. The basic kinds of image pre-processing comprise brightness transformations, geometric transformations, image filtering and Fourier transforms. Pixel brightness includes contrast adjustment or contrast enhancement. Brightness transformation involves brightness corrections and gray scale transformation. Further, most common pixel brightness transformations include Gamma correction, sigmoid stretching and histogram equalization. In geometric transforms, the geometric distortion is excluded. Two basic types of geometric transformations comprise spatial transformation and grey level interpolation. Image filtering comprises low pass filtering, high pass filtering, directional filtering, and Laplacian filtering. Fourier transforms comprises the Discrete Fourier Transforms (DFT).

1.2.3 Segmentation

Segmenting the image is the phenomenon in which the image is split-up into sub-groups, also called as the objects. Doing this not only brings down the intricacy of the image, but also makes the analysis of the image easier. A number of algorithms are used involving segmentation that can separate and categorize a defined set of pixels together from the input image. In the said process, in actuality, we are allocating labels to pixels and also these pixels with the same label are included in a category where they have something common. With the aid of these labels, lines or boundaries can be framed and hence separation of the most significant objects can be done from the insignificant parts of the image. Considering the above point from the machine learning perspective, these associated labels can be later on utilized for both supervised and unsupervised learning which aids in analysis of an expansive range of business issues. There are several types of image segmentation techniques that comprise threshold method, edge based segmentation, region based segmentation, clustering based segmentation, watershed based segmentation and Artificial Neural Network (ANN) based segmentation.

1.2.4 Feature Extraction and training

A feature is considered as a section of details regarding the content in an image which is mostly required to check if a particular portion of the image is having specific characteristics or properties. It is generally a part of information which is appropriate to work out the calculations related to specific application. Features in image processing and pattern recognition have very advanced set of features. The notion of features is very extensive but the selection of features in computer vision model is absolutely reliant on the given problem. A feature is also considered as an interesting portion of an image. The aim is to search for interesting patterns in the image that are feasibly distinctive to a specific class which will further aid the model to discriminate between several classes. The strategy is referred to as the feature extraction if features are represented in terms of confined neighborhood. The different types of features may comprise edges, corners, blobs, ridges, etc. A model acquiring knowledge regarding the features from the dataset is known as model training.

1.2.5 Classification

The last step in image processing is responsible to designate the identified objects into pre-specified labels or classes with the aid of appropriate classification algorithms. This step also contrasts the patterns of the image with the patterns of the target. Classification can be broadly categorized as supervised classification and unsupervised classification. In supervised classification, the notion is that few representative pixels of a particular class from the image are selected which are later on used to administer the model to utilize these representative pixels as references in order to categorize all other pixels in the image. Specific criteria can be set by the user as to how much similarity should be there between the pixels in order to group them together. Criteria can be set over the total number of labels or classes as well. Once the feature extraction has been performed or characteristics have been obtained, the classification of the image is performed by determining the impression for each pixel and then coming to the conclusion that to which particular label it resembles the most.

Predictive models can be developed using supervised classification that comprises classification and regression techniques. The algorithms used in supervised learning are support vector machines (SVM), k nearest neighbor (KNN), Naïve Bayes (NP), decision trees (DT), random forests (RF), logistic regression, linear regression and neural networks. In unsupervised classification, there is no need to specify the sample classes in order to determine the outcomes that is assembling the pixels with familiar characteristics. Different techniques are used in computer vision in order to decide which pixels have similarities and then assemble them into classes. Number of classes and the algorithm to be used can be specified by the user.

1.3 MOTIVATION

The leaves of a plant are a vulnerable element. The classification of agricultural harvests is dynamically evaluated. The texture and color of the leaves are the most prominent visual properties. As a result, evaluating agricultural produce, boosting market value, and satisfying quality standards necessitate the classification of leaf disease. It is also beneficial to identify and take further steps to prevent illness spread.

If identification and categorization are done using physical approaches, the process will be excessively sluggish. As a result, we require the assistance of professionals. It may be failure at times, or experts may not be approachable. Disease is classified by the farmers based on color, size, and other factors. The endeavour will be error-free and speedier if these attributes are recorded into a system utilizing appropriate programs. Farmers will benefit from this technology since it will inform them at the correct time, even before disease spreads across a big area. So, our prime focus will be upon the following points:

- To contribute towards research in the field of leaf disease detection to help the farmers
- To contribute towards research in India since not enough work has been done in the area of plant leaf disease detection
- Handling small-sized dataset consisting of diseased leaf images.
- Use of machine learning algorithms

1.4 PROBLEM STATEMENT

Early and accurate detection of plant diseases has a positive impact on crop productivity and quality. Farmers may struggle to recognise diseases in plants by viewing disease affected leaves due to the production of a vast variety of crop items. However, in underdeveloped countries' rural areas, visual inspection is still the major method of illness detection. Therefore, it necessitates expert monitoring on a regular basis. Farmers in rural places may have to travel a long distance to visit an expert, which is both time-consuming and costly. The proposed semi-automated system intends to assist the farmers in identifying leaf diseases with least human intervention. By applying image processing techniques, the system incorporates the use of robust hybridization of feature extraction and classification techniques. The focus is on convolution based Law's mask features and ensemble classification.

1.5 RESEARCH OBJECTIVES

- Implement Law's textural mask method for the feature extraction of plants for disease detection
- Design ensemble classification method for the classification of plant features for the disease detection
- Implement designed approach and compare with existing techniques in terms of accuracy, precision and recall.

1.6 CONTRIBUTIONS TO THE THESIS

- The work is an attempt to improve the semi-automated system for plant leaf disease identification.
- It will help the farmers in timely identification of plant leaf diseases and hence the farmer will be able to prevent the disease to spread in the entire field.
- It will reduce the manual expertise and the cost required for detecting plant leaf diseases.

- It will be less error-prone as compared to human expertise.

1.7 THESIS ORGANISATION

Chapter 2 conducts a thorough assessment of the literature on various segmentation, feature extraction and classification algorithms for plant leaf disease diagnosis. It also throws light on the research gaps found in the state-of-the art publications, and the benchmark dataset utilized in our work.

Chapter 3 suggests the usage of an optimization based segmentation and Law's mask feature extraction with a combination of other feature extraction techniques. Various combinations have been tested and the best approach has been identified. Support Vector Machine (SVM) is utilized as a classifier to classify the leaf diseases on the basis of various performance metrics.

In Chapter 4, we extract texture based features using the standard feature reduction techniques. The results are compared with the best identifies approach from Chapter 3.

Chapter 5 focuses on utilization of numerous machine learning classifiers to construct an ensemble classifier. Certain techniques are hybridized and best approach is identified A deep learning based Convolutional Neural Network approach has also been tested.

Chapter 6 is the extension of Chapter 5. The best identified approach from Chapter 5 has been hybridized with Gabor filter and the results are compared. It also focuses on a comparative study among the final proposed approach and the existing approaches proposed in the recent years.

In Chapter 7, architecture of semi-automated approach is proposed that predicts disease based on the user's input of a leaf image.

Chapter 8 concludes our thesis and gives an insight into the future scope.

CHAPTER 2

LITERATURE SURVEY

Diseases affect production of plants, limiting its growth and lowering the amount and quality of the plant. Image processing is the most effective method for detecting and diagnosing diseases. Various publications presenting plant leaf disease diagnosis are presented and discussed in this chapter. We have conducted a thorough assessment of the literature on various segmentation, feature extraction and classification algorithms for plant disease identification. The chapter also discusses the research gaps found in the existing literature. A benchmark dataset used in our work has been discussed in detail.

2.1 SEGMENTATION

Grouping the image into numerous fragments is referred to as segmentation. It can be, apparently considered as the sub process of pre-processing phase. Disconnectedness (discontinuity) and affinity (similarity) are the two characteristics algorithms based on segmentation. There are two approaches being used in segmentation. Similarity detection is the region approach and discontinuity is the boundary approach. The first approach is reliant on identifying similar pixels in the image. Segmentation algorithms such as clustering utilize this approach of similarity identification on an unspecified set of features. The latter approach is reliant on the discontinuity approach. Algorithms such as line detection, edge detection or point detection are reliant on this approach.

2.1.1 Threshold method

The threshold method is undoubtedly one of the robust techniques to detect the desired objects in an image. The pixel's intensity is compared with the threshold value and hence the pixels in the image get separated based on the intensity. This method is quite helpful when the foreground objects in the image are supposed to have higher intensity as compared to the objects in the background (probably undesired components in the image). The threshold value can either be kept constant or can be changed dynamically on the basis of properties of the image and therefore better results can be achieved. Thresholding can be further classified as simple thresholding, Otsu's binarization and adaptive thresholding.

- **Simple thresholding** - In simple thresholding, the pixels in the image are replaced by either white or black. Provided that the pixel intensity ($I_{i,j}$) at a location (i,j) is greater than the threshold (T), then it is replaced with white, otherwise black if vice versa. This method has been utilized for crop identification in [37] on the basis of Hue Saturation Value (HSV) segmentation. Color pixels have been utilized in order to perform threshold segmentation [20].
- **Otsu method** - Otsu method is useful when the histogram of an image has two peaks, that is foreground and background. Hence, in order to determine the threshold value in Otsu method, a value at the center of the peaks can be assumed as the threshold value. This method cannot be used in case an image histogram has multiple peaks. This method can be used for removal of undesired colors from a document. Otsu method [24] has been utilized on H component of Hue Intensity Saturation (HIS) color space for performing image segmentation.
- **Adaptive thresholding** - Sometimes finding a global threshold value may not be a nice idea as it may not prove to be good in all the situations such as if the image is comprised of foreground and background with different lighting conditions. In adaptive method, the threshold value can be changed for several parts in the image. Here, the image is divided into a number of small-sized parts and then computation of threshold for those parts in the image is performed. Therefore, multiple threshold values are achieved. So, we can conclude better results for images with differing illumination. The threshold values are computed automatically by the algorithm.

2.1.2 Edge based segmentation

Edge detection is the phenomenon of tracking down the edges in an image which is a significant stride towards realizing the image features. It is accepted that edges comprise significant features and contains meaningful information. It remarkably diminishes the size of the image that is processed and sieves out the information which is possibly not that important or significant, conserving only the important characteristics of the image. Edge detection algorithms can be categorized as gradient based methods and gray histograms. In these algorithms, operators such as Sobel operator, Canny [97] and Robert's variable can be used. These operators help in detecting the discontinuities of the edge and therefore trace the edge boundaries. The ultimate objective is to achieve at the minimum a partial segmentation where the native edges can be grouped into a new binary image where only those edge series are present that complement with the desired existing objects. Canny filter has been employed for locating wide range of edges in images. Peak Signal-to-Noise Ratio (PSNR) for canny filter was found to be the lowest while Mean-squared Error (MSE) was found to be the highest. An improved version of Canny filter has been used [101] for de-noising the image which indicates that edge information can be extracted in more accurate way.

2.1.3 Region based segmentation

The algorithms based on region segmentation create the chunks by splitting up the image. The image is divided based on affinity between the components. The components are actually represented by the pixels themselves. These kinds of segmentation algorithms initially focus on finding few seed points. These seed points may be either tiny parts or huge portions in the input image. After this, specific approaches are applied that can append more pixels to seed points or further reduce or contract the seed point to tiny portions and integrate with other tiny seed points. Therefore, the basic approach comprises region growing and region splitting and merging.

- **Region growing** – This method is a bottom up approach. Here, we initialize with a tiny set of pixels and begin gathering or constantly joining it on the basis of pre-selected affinity or similarity constraints. If a match is found in the contiguous pixels, then they will be joined to the commencing seed pixel, therefore expanding the size of the region. When saturation is achieved, that region cannot be expanded further, so the algorithm has to choose another seed pixel that may or may not belong to any region that presently exists and begins the process again. Effective segmentation results are achieved in region growing methods that communicate well to the perceived edges. In contrast to this, if we let a region expand entirely prior to testing other seeds, that typically biases the segmentation in favor or regions that are segmented first, so in order to avoid this situation, maximum algorithms start with the user inputs of affinity first and hence no sole region is granted to control and expand. Therefore, numerous regions are permitted to expand concurrently. Region growing algorithms resemble other pixel based algorithms such as thresholding. But the prime difference between these two is that thresholding draws out a huge region on the basis of similarity of pixels whereas region growing pixels draws out only the neighboring pixels. Region growing based segmentation algorithms are highly recommended for images consisting of noise wherein it becomes cumbersome to identify the edges.

- **Region splitting and merging** – These algorithms use two techniques performed together in union. One is region splitting and the other one is region merging in order to segment the image. Splitting comprises constantly splitting up the image into regions that have like properties or characteristics. Merging involves joining the adjoining regions which are a bit alike to each other. In contrast to region growing algorithms, these algorithms take into account the whole input image as the area of interest. Then it uses pre-selected similarity restrictions and fetches all the pixels satisfying the criteria. Therefore it is a "divide and conquer" approach as compared to the region growing approach. The said process is just first half of the process. Later on, the split process is performed and we obtain a number of likewise marked regions spread across the image pixels. This implies that the ultimate segmentation consists of dissipated clusters of adjoining regions which have like properties. In order to fulfill the process, merging is required to be performed. The adjacent regions are compared based on the degree of similarity and merging is performed.

2.1.4 Clustering based segmentation

In contrast to classification algorithms, the clustering based algorithms are unsupervised. Here, the user is not accessible with the pre-selected group of features or classes. The algorithms based on clustering aid in getting the elementary concealed information from the data such as clusters and groups which are generally not known from the analytical perspective. These techniques are meant to divide the image into bundles or disunited groups of pixels with like characteristics. These pixels are grouped into bundles or clusters in such a way that the elements in the same cluster are more alike to each other in contrast to other clusters. The different approaches to clustering algorithms comprise K-means clustering and Fuzzy-C means clustering (FCM). K-means clustering is a selected and well accepted method as it is simple and computationally efficient. A variation of K-means clustering includes the minimization of the number of iterations required in K-means clustering algorithm. In contrast to K-means clustering, another algorithm known as FCM permits the pixel points to be associated with numerous classes having varied degrees of associativity or membership. A variation of FCM algorithm involves optimization of the steady processing time.

- **K-means clustering** – It [53] is one among the easiest unsupervised learning algorithms that can resolve problems related to clustering. Basically is comprises two major stages – determining the ‘k’ centroid and allocation of points to the clusters. The classification of a provided image is executed via a specific count of clusters that are a priori. The algorithm begins at this stage where division of the image space is done into k pixels constituting k group centroids. Then all the pixels are allocated to the group on the basis of its length or distance from the cluster. The distance considered here is the Euclidean distance. After each and every pixel has been allocated to every cluster, the centroids are moved and therefore, are reallocated. The said steps are rerun till the centroids can no more shift. When this algorithm obtains convergence, k groups are achieved in the areas within the image where the constituting pixels reflect some level of likeness. Minimization of an objective function is required in K-means clustering algorithm (refer to Fig 2.1). In [30], ‘a’ and ‘b’ components are used with the aid of K-means after converting to CIELAB. One of the salient features of K-means clustering is that the clusters and the components are absolutely mutually exclusive.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i^{(j)} - c_j||^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.1)$$

In equation 2.1, J is the objective function, k denotes the number of clusters, n denotes number of classes, i indicates the case, c represents the centroid for cluster j.

- **FCM** – This algorithm permits the pixels or data points to be associated to numerous or multiple clusters. Alternatively, a bunch of pixels can be part of more than one cluster but the levels of membership can vary per group. An optimization function is involved in an FCM algorithm and the minimization of this function concludes the convergence of algorithm. When this algorithm obtains convergence, C groups are achieved in the areas within the image where the constituting pixels reflect some level of likeness but they will also have a definite degree of membership or association with the rest of the groups as well. The typical Fuzzy C-means is a type of clustering that permits a portion of the image to be comprised into more than one cluster or multiple clusters. An objective function is required to be minimized in this case.

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} |x_{ij} - C_j|^2, \quad 1 <= m < \infty \ldots\ldots\ldots(2.2)$$

Here, x is the sample set, n is the sample number, C is the cluster center, m is the fuzziness parameter or weighting exponent.

Calculation of gray distance [46] is done between the pixels and cluster center. This determines the fuzzy associativity value of the pixel and therefore aids in the ultimate result of segmentation. A low fuzzy associativity is represented by a high gray distance and a high fuzzy associativity is represented by a low gray distance. In [46], FCM involves the use of spatial information that includes weighted gray and spatial characteristics in order to enhance the objective function.

2.1.5 Watershed based methods

Watershed is actually an alteration employed on a grayscale image. The name of the algorithm is analogical to a topographical or terrestrial watershed that splits up adjoining drainage basins. The image is handled in this algorithm as if it is a topographic map. It is a kind of region based method, also called as the ridge approach and is in accordance with the notion of topological interpretation. Let's assume a metaphor in which we have a geographical or topological landscape having cliffs and valleys for various components in the image. The slants and the altitudes of the provided topography are clearly quantified by the respective pixels' gray values, known as the gradient magnitude. On the basis of this 3D depiction, which is generally in accordance with landscape of the earth, the image is decomposed by the watershed transform into regions which are known as "catchment basins". For each local minimum, a catchment basin includes all pixels whose path of steepest descent of gray values ends at this minimum. In other words, this algorithm represents pixels as the "local topography" or elevation, most of the time starting itself with the markers defined by the user. Then, the minima points called the basins are defined by the algorithm. The basins are separated from each other by the watersheds established here. Therefore, the picture gets broken up as pixels belonging to each such region are obtained. For segmentation purpose, watershed background [25] is utilized to segment the objects where the watershed lines are represented by edges, nodes and lines. A number of definitions exist for this algorithm. Watershed by flooding, watershed by topographic distance, watershed by the drop of water principle, inter-pixel watershed are a few variations. In the following example, two conjoining or intersecting circles are required to be split up. In order to do the same, an image is determined which is the distance to the background. This distance's maxima are selected as the markers and the two circles are split up by the flooding of basins across a watershed line. Refer to Fig 2.1.

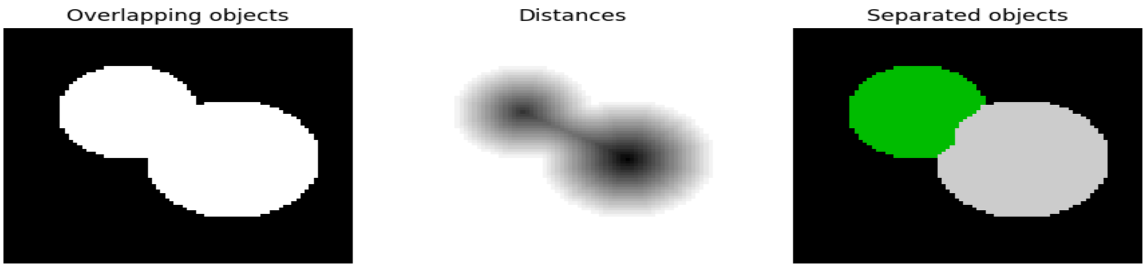


Fig 2.1 Watershed segmentation [107]

2.2 FEATURE EXTRACTION

The parameters that aid in the description of an object are referred to as features. Features can basically be categorized as global features and local features. The global features can be further categorized as color histograms, texture histograms and color layout of entire image. The local features can be categorized as texture, color and shape features for the divided portions as a part of the segmentation process. These features after being extracted from the images are later on utilized for comparing the images to perform matching and retrieval. A feature extraction method Pyramid Histogram of Gradients (PHOG) [53] can be useful for both color based and texture based feature extraction. Texture and color features can also be extracted using Color co-occurrence method [45]. For components, that is, H S I, three color co-occurrences were produced.

2.2.1 Color features

One of the ultimate and elementary visual features of an image comprises color as the human visual system is receptive to colors. Color is a rudimentary characteristic of the image content. The color features facilitate the humans to identify maximum images and contents or objects in an image. Histograms can represent these images by showing the amount of pixels of each color amidst a range. Most of the time, in this case a histogram is achieved that separates the range of data into identical bins. Each bin is determined for the total number of pixels which have the identical or same color values. So, color features are usually utilized for image retrieval. Numerous approaches have been adduced to retrieve images based on the color identicalness of colors. Most of these approaches follow the same notion. Every image stored in the database is scrutinized to evaluate its feature. Global Color Histogram (GCH) facilitates in exhibiting the images by their histograms and the identicalness between the two images is evaluated by the distance between their color histogram. This method is not that helpful in exhibiting the image appropriately. Moreover, differences in intensity and distortion of colors are a drawback in this case as GCH is sensitive to all these. There is another approach known as Local Color Histograms (LCH) that splits up the image into blocks and then determines the histogram for every block independently. So, the image is exhibited by these histograms. In order to perform the comparison between the two images, every block from the first image is matched with another block from the second image at the same position. The sum of all distances will be the distance between the two images. This approach aids in exhibiting the image more in detail and hence empowers us to compare between the image regions. Identification of infectious marks on the leaves of plants can be performed with the aid of color features without difficulty because color is the readily available detail exhibited by the image. Color spaces aid in determining the color histograms. Three color channels (Red, Green, Blue) [25] are the main components. The color histograms are susceptible to differences in lighting, so they are transformed to HSV color space in order to separate intensity and chromaticity. Comprehensive color feature [40], [50], [32] identifies disease marks with the aid of Excess Green Index and Excess Red Index and with even illumination and is very advantageous. The image is transformed into HSV and L*a*b* color space. This is performed in order to extract H and b* components. Hence (ExR, H, b*) may be utilized which is absolutely a distribution of Comprehensive Color Feature (CCF). To separate the plant, weed and soil, HSV [22] color space has been utilized. By matching the values of hue and plant pixels, the dissociation of the background is performed. [37] uses HSV color space for performing segmentation for making the approach more powerful and its comparison was performed with [23]. Transformation of Red Green Blue (RGB) image to L*a*b* color space [52] is near to human judgement. In [15], 2d xy color histogram aids in training SVM. Instead of using RGB color space, CIE color space has been utilized. H color channel of HSV and 'a' color channel of L*a*b* [27] are exploited for differentiating between symptoms of the infections and the tissues of the plants. 14 distinct color spaces [103] have been explored and extraction of 4 features has been performed from each color channel, making up to total of 172 features.

2.2.2 Texture features

Texture is referred to as a feature that is utilized to split up the regions of interest and then categorize or classify those regions. The information imparted by texture is spatial in nature in the form of positioning colors or intensities in an image. Texture is actually defined by the spatial composition of levels of intensity in adjacent regions.

Texture is actually a recurrent pattern of neighborhood variations in the intensity of image. A point cannot have a texture. A texture can be expressed as fine, coarse, grained, smooth, etc. These features can be normally located in the structure and tone of the texture. Tone is built on the pixel intensity whereas a structure is constituted by the spatial relationship among pixels. Two important terms are associated with texture analysis – texture segmentation and texture classification. Texture segmentation is referred to automatically computing the edges among different texture portions in an image. Texture classification refers to detecting a provided textured portion from a given group of classes of texture. Every region may have a unique feature property. An example of this comprises statistical method such as GLCM that can further compute features such as contrast, entropy, correlation, etc. These statistical approaches are advantageous when the texture primitives are tiny which can be further termed as microtextures. Another type of approach comprises the Law's texture features. It is actually a group of convolution kernels which are initially determined from a set of vectors L5, E5, S5, R5 and W5. L5 refers to level, E5 refers to edge, S5 refers to spot, R5 refers to ripple and W5 refers to wave. A texture in a leaf is represented by the parallel lines. For analyzing text in images, Gabor filter [25] was used which comprises bandpass and linear filters. A Gabor filter is actually a product of a Gaussian Envelope function and a complex oscillation. So, it computes the frequency part in a specific direction in a specific region in an image. In [25], Gabor based Haralick feature extraction filter has been utilized for extracting texture features and identifying edges. It used Gabor wavelets for performing feature extraction. Orientation, standard deviation, and radial center frequency are the parameters used in this case. These are generally utilized for signal processing, therefore the Gabor filter size is required to be minimized for conquering the problems related to dimensionality. In traditional Haralick approach, extraction of eight features is performed at an angle of 45 degrees of rotation. A variation of this proposes feature extraction for every 30 degrees rotation. As a result, extraction of 13 different features is possible with the said variation. Hence, extraction of leaf features becomes easier. Grey level color co-occurrence method (GLCM) [30], [32] is one of the earliest approaches and is utilized to determine spatial relationship between the couple of pixels. It is also referred to as gray-level spatial dependence matrix. The determination of how frequently the couple of pixels having specific values and in a specified spatial relationship take place in an image. The prominent features taken into account comprise contrast, correlation, energy, entropy and homogeneity. In [6], an approach Deterministic Tourist Walk (DTW) has been adduced on the basis of histograms in order to execute texture feature extraction in images of infectious plant leaves. Traversal of the image is performed by a traveller on a provided scale based on this technique. Refer to Fig 2.2.

Fig 2.2 Texture features [108]

2.2.3 Law's Mask Feature Extraction

Kenneth Ivan Laws proposed the Laws' masks strategy of extracting features in 1980, with the fundamental idea being to filter images with specified masks created from a collection of one-dimensional kernel vectors in order to evaluate texture qualities. It has been successfully used for wood defects categorization [42], mammogram classification [17] and analyzing breast lesions [29]. To calculate the texture qualities of photographs, the Laws masks compare the pixel vicinity to a collection of standardized masks. The two-dimensional convolution kernels commonly used for texture differentiation are created from the following set of three- and five-dimensional convolution kernels : L3 = [1 2 1], E3 = [1 0 −1], S3 = [1 −2 1], L5 =[1 4 6 4 1], E5 = [1 −2 0 2 1], S5 = [−1 0 2 0 −1], W5 =[−1 2 0 −2 1], R5 =[1 −4 6 −4 1]. L is level detection, E is edge detection, S is spot detection, R is ripple detection. Table 2.1 shows comparison chart for state-of-the-art papers using Law's texture mask.

Table 2.1 Comparison chart for Law's texture mask

Paper	Year of Publication	Technique	Main findings
[4]	2008	Law's Texture Energy Measure	Was found to be efficient for bone texture analysis
[17]	2015	Law's Texture Energy Measure	Law's - 93.90% accuracy GLCM – 72.20% accuracy
[29]	2016	Law's Texture Energy Measure	Law's - 97.4% accuracy
[42]	2017	Law's Texture Energy Measure	90.4% overall accuracy

2.2.4 Grey Level Co-occurrence Matrix

The grey level co-occurrence matrix (GLCM) technique [18] [14], also known as the spatial grey level dependence matrix (SGLDM) approach, considers second order statistics: pairings of pixels in specific spatial relationships are investigated. Co-occurrence matrices are employed for this purpose. The GLCM is utilized to extract texture from an image by performing a gray-level comparison between two pixels. Within a picture, the GLCM produces a joint distribution of grey level pairings of contiguous pixels. Haralick proposed statistics formulae for characterizing picture textures that may be determined from the co-occurrence matrix. It's a statistical method of inferring image texture organization by statistically selecting the pattern of grey-levels in respect towards other grey levels. The normalized co-occurrence is largely represented by weighted averages. By multiplying the elements of a grid by a weighted mean multiplier, the relative importance of the item can be expressed.

The above matrix has a dimension of N, which describes the amount of grey levels in the picture. The matrix's element [i, j] is created by determining the probability with which a singular pixel with value i is contiguous to a pixel with value j, then dividing the matrix by the overall number of these assessments. As a result, each result in the matrix represents the likelihood that a pixel with value i will be discovered contiguous to a pixel with value j. The equation for the specified co-occurrence matrix is calculated from the Image (i, j).

2.2.5 Local Binary Pattern (LBP)

The Local Binary Pattern (LBP) [35] is a straightforward but powerful texture operator that labels pixels in a photo by partitioning the neighborhood of each pixel and converting the result to a binary value. The LBP system can be viewed as a fusion of texture assessment, usually divergent statistical and morphological interpretations. The LBP operator's randomness alongside monotonic grey level changes, for example, brightness variances is perhaps its most important quality in real-world applications. Another key feature is its operational ease, which allows it to evaluate photos in perplexing real-time scenarios. The fundamental LBP operator was based on the hypothesis that texture has two balanced aspects: a pattern and its strength, which are both present locally. The operator uses the center value as a threshold and works in a 3x3 neighborhood. Multiplying the thresholded values with the values provided by the relevant pixels and totalling the result yields an LBP code. Because the neighbourhood consists of 8 pixels, the comparative grey values of the epicentre and the pixels in the locale can be used to generate a total of 28 = 256 different labels. By deducting the mean of the grey levels below the centre pixel from the mean of the grey levels above (or equal to) the centre pixel, the contrast measure (C) is obtained. The amount of contrast is initialised to 0 if the eight thresholded neighbours of the central pixel have the same value (0 or 1). LBP code variations are utilized as clustering or segmentation characteristics. The LBP operator transforms a picture into a group or photograph of integer labels that describe the image's small-scale characteristics. These labels or their characteristics, one of the most common of which is the distribution, are then applied to the image for additional analysis. The operator's most often used variants are for grayscale still photos, but it has been expanded to include color (multi-channel) pictures, movies, and volumetric data as well.

2.2.6 Gabor Filter

For texture analysis, the Gabor filter [11], commonly known as the linear filter, is used. It's mostly utilized for detecting frequency content in unusual orientations in a picture that are confined in a region. The Gabor filters' depiction is remarkably similar to the human visual system. Four scales and six orientations are used to extract Gabor features [52]. Gabor features along with color features achieved the best results. For disease detection, the suggested system in [13] employs a basic Gabor Wavelet texture method. Each image has been subjected to the Gabor filter function:

$$g(x, y;\ \Upsilon, \theta, \psi, \sigma, \lambda) = \exp\left(-\frac{x'^2 + \Upsilon^2 y'^2}{2\sigma^2}\right)\exp\left(i\left(2\pi \frac{x'}{\lambda} + \psi\right)\right) \ldots\ldots\ldots\ldots (2.3)$$

Here, λ represents the wavelength of the sinusoidal factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, σ is the standard deviation of the Gaussian envelope and   is the spatial aspect ratio.

2.2.7 Scale Invarient Feature Transform (SIFT)

SIFT [49] is used to represent local characteristics using feature vectors. For picture based recognition, it employs an image descriptor. It utilized picture key points, which were then combined with descriptions to recognize things.

2.2.8 Shape features

The visual features in images perceptually belong to shape features. Shape features can be categorized on the basis of properties of boundaries properties of region. The shape feature extraction approaches can be categorized as boundary based and region based. A shape descriptor is a group of numbers which are constructed to constitute a provided shape feature. A descriptor seeks to evaluate the shape features in ways that are in accordance with human perception. These descriptors are generally in the form of vectors. There are two classifications for these descriptors – contour based methods and region based methods. The feature extraction approaches on the basis of boundary comprises simple feature descriptors such as perimeter, diameter, eccentricity, curvature, etc. Other descriptors for boundary feature extraction comprise fourier descriptors and statistical moments. The feature extraction approaches on the basis of region comprises regional area, roundness, regional focus and topological descriptors. Both boundary and region based descriptors are informative and replaceable, means that one can be utilized as a foundation for calculating the other. For object identification and categorization, shape representation is a significant issue. The representation of shape is provided in respect of association between such objects. These characteristics conform to properties of the objects' location, shape and size. Every image stocked in the database is exercised upon to achieve the shape features. These shape features are later on utilized by various shape representation approaches or techniques in order to arrange the helpful facts and details in index structures for systematic retrieval. For instance, the connected edges or borders express the properties of the shape object. Hence, the shapes can be worked upon to achieve their shape borders. Then, these shape borders are automatically broken down into a group of boundary points which are generally utilized in machine vision approaches for shape recognition. A provided set of features may give appropriate results for specific applications. But, they may not do the same in case of other set of applications. Hence, only those features should be extracted by the shape representation that experts feel is appropriate for that particular application. The shape features must meet certain conditions. First of all they must acquire good differentiating properties. The shape representation should be able to estimate the objects and simultaneously continue their ultimate and significant features. Shape features should not be affected by transformations such as scaling, translation and rotation. They need to be powerful and abstract from any kind of malformation. They must allow identification of intuitively identical objects which are not mathematically similar. Also, they should be easy to derive. The accuracy of result of any shape feature extraction approach is dependent on the pre-segmentation effect. The utilization of shape of an object is a challenging issue in generating a robust Content-Based Image Retrieval (CBIR). The shape plays a crucial part to find identical objects in the image. Shape features are barely evolved as compared to color and texture features due to the intrinsic property to represent images.

2.3 CLASSIFICATION

In statistics and machine learning, it is a supervised learning scheme in which learning is done by the computer program from the data provided to it and then builds up observations and categorization. The aim of a classification procedure is to designate a provided set of data to classes or labels. Classification is applicable to both structured as well as unstructured data. This phenomenon begins with identifying the class of the provided data points. The classes are, most of the time, mentioned as labels, categories or targets. The mainstream classification problems comprise face detection, voice recognition, handwriting analysis, document classification, etc. The classification can be further categorized as binary or multi-class. A binary classification has only two practicable outcomes while a multi-class classification consists of more than two outcomes.

2.3.1 Support Vector Machines (SVM)

SVM is a linear model for problems dealing with both classification and regression. But it can work and find solutions for both linear and non-linear problems. It is a classifier that constitutes the training data as points in the space split up into classes by a space which is as broad as possible. In other words, it searches for an isolating line between the data of both classes. This algorithm accepts the data as an input and then produces a line which isolates or separates those classes, in case if feasible. After that, the latest points are appended in the space by identifying the class they should belong to and the space that they will belong to. SVM [45], [15] is a supervised learning approach and can be contemplated as a hyperplane which detaches the data points of classes for solving classification and regression problems. Largest hyperplane pinpoints good separation that is obtained by largest distance to the nearest training data point. The advantage of SVM is that it is very effectual in high dimensional spaces. Let's take the following example in which we have a distribution of green circles and blue squares. We are required to categorize green circles from blue rectangles. Practically thinking, there can be infinite number of lines that have the potential to split up the provided two set of classes. But, which one of them is the best? Points that are nearest to the line are searched in both the classes. These points are referred to as support vectors and the distance is denoted by margin. The intent is to enlarge the margin to the maximum possible extent. The hyperplane achieved with the maximum margin is denoted by optimal hyperplane. Refer to Fig 2.3 and 2.4 to understand the concept of hyperplane finding.

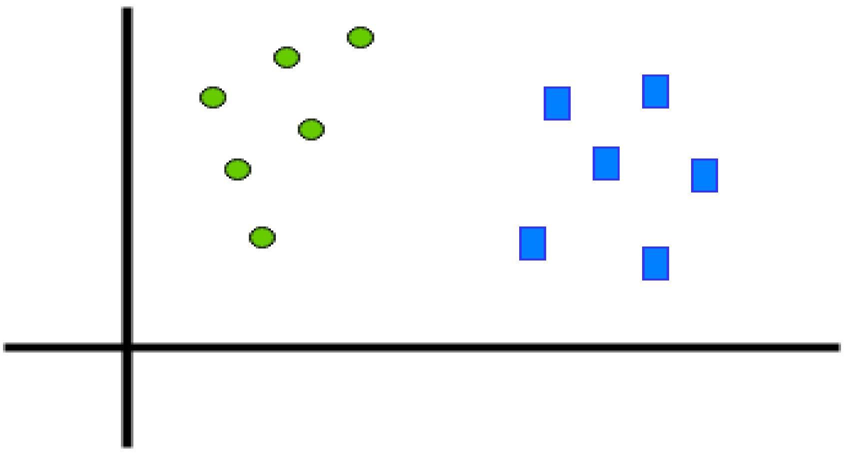


Fig 2.3 Search a hyperplane that results into separation of green and blue categories

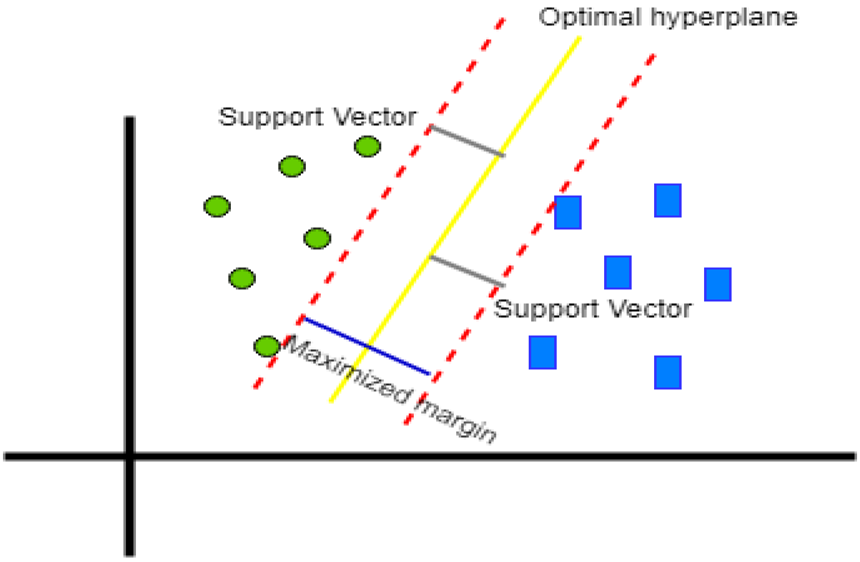


Fig 2.4 An ideal hyperplane using the SVM

Fuzzy Relevance Vector Machine (FRVM) is a refined form of SVM. In this, the rules are built on the concept of fuzzy logic. This approach facilitated for 60 different leaf classes [30]. FRVM rules are established as R1, R2, …Rn where R1, R2, Rn denote the scale of training features. The key power of FRVM is that it minimizes complication (because of RVM).

2.3.2 K nearest neighbor

Abbreviated as KNN [25], this algorithm is easily practicable and is a supervised learning approach and can be utilized for resolving both classification and regression related problems. This algorithm supposes that identical objects happen in immediate vicinity. It is commonly utilized when there is less or no prior cognizance regarding the arrangement of data. First of all, the data is required to be loaded. Then, the value of k is initialized. Iteration is performed, beginning from 1 to the total count of training data points. Later on, the distance (generally Euclidean distance) is computed between the test data and each line of the training data. The computed distances are then ordered in ascending order of on the basis of the values of distance. Then, top k rows are obtained from the ordered array. Highly recurrent class of these rows is obtained and therefore the predicted class is returned. Consider the following example that shows the distribution of green circles (GC) and blue squares (BS) as in Fig 2.5.

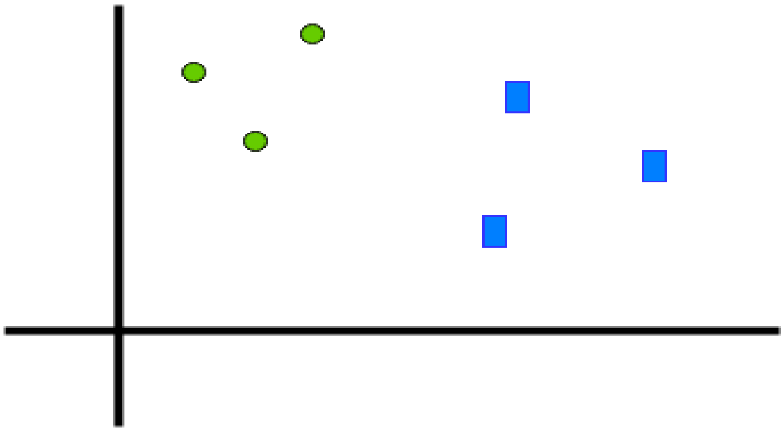


Fig 2.5 Distribution of green circles (GC) and blue squares (BS)

The motive is to obtain the class for the red star (RS) which can be any one out the two given classes. Now, we intend to find out the nearest neighbor, k. Let us assume k as 3. Therefore, a circle is drawn big enough to accumulate only three neighbors (points) in the plane. The three nearest points to RS are all GC. Hence, it can be concluded that RS belongs to class GC as in Fig 2.6.

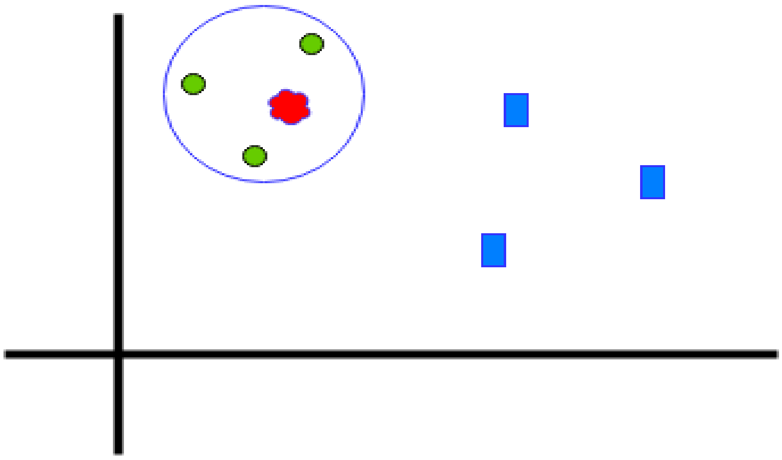


Fig 2.6 RS belongs to class GC

2.3.3 Logistic Regression

Logistic regression is a significant aspect of regression analysis. Before we know about logistic regression, let's throw some light on regression analysis. It is basically a predictive modeling approach that is utilized to search association between a dependent variable and an independent variable. A broad classification of regression analysis comprises - Linear regression and Logistic regression. Logistic regression facilitates in the prediction of binary outcomes, means only two classes are feasible. By default, logistic regression is limited to only binary class classification problem which is also referred to as binomial logistic regression. But, it can be used for multi-class classification as well. It is also referred to as multinomial logistic regression. Logistic regression utilizes an equation, similar to a linear regression. The values of the input are joined in a linear way with the help coefficient values or weights. Following is the equation of logistic regression.

$$\boldsymbol{y = e^{b_0 + b_1 * x} / (1 + e\hat{}(b_0 + b_1 * x))} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.4)$$

where, the predicted outcome is denoted by y,
intercept or bias is denoted by b_0,
coefficient for single input value is denoted by b_1

Logistic regression uses a sigmoid function. The function is also referred to as logistic function, which is actually an S shaped curve. It can accept any real value between 0 and 1. It can be defined as

$$\boldsymbol{\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.5)$$

S curve is plotted as in Fig 2.7.

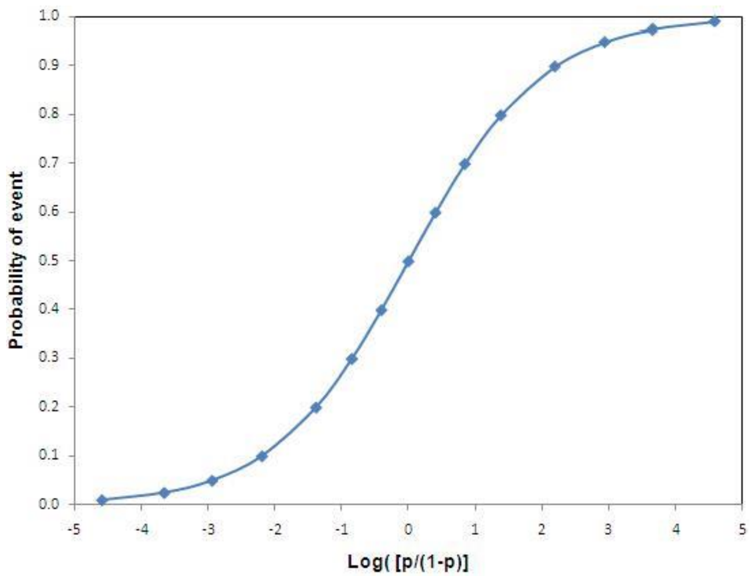


Fig 2.7 S curve [109]

Positive infinity curve outputs the y predicted value as 1. Negative infinity curve outputs the y predicted value as 0. Logistic regression ensemble with deep learning approach [85] obtained the highest accuracy of 97.59% for identifying four peanut leaf diseases.

- **Binomial logistic regression** – Predicting the association between a dependent variable and an independent variable is referred to as a binomial logistic regression. Here, the dependent variable is of type binary.
- **Multinomial logistic regression –** In this case, there is single dependent variable and is categorical in nature but the outcomes are more than two.
- **Ordinal logistic regression** – The dependent variable is ordered here.

2.3.4 Naïve Bayes

Naïve Bayes [83] is actually a group of supervised learning algorithms that follow the same principle. There is a naïve assumption that between every couple of features, a conditional independence is must. In other words, it indicates that existence of a specific feature in a class is not related to the existence of any other feature. This model is very helpful in case of large datasets. This classifier is built on the Bayes theorem. Following is the Bayes theorem.

$$P(A/B) = \frac{P(B|A)P(A)}{P(B)} \quad \text{.......................................(2.6)}$$

Provided B has already happened, we can compute the probability of A’s occurring. Hence, hypothesis is denoted by A and evidence is denoted by B. There is supposition in this case. It is that the features are considered independent. Independence among the predictors is a strong assumption. Posterior probability P(c|x) can be computed from (c), P(x) and P(x|c) through following equation:

Bayes’ theorem follows the following relationship, provided class variable and dependent feature vector through

$$\mathbf{P(y \mid x1, \ldots \ldots, xn)} = \frac{P(y)P(x1,\ldots.xn \mid y)}{P(x1,\ldots.,xn)} \quad \text{...................... (2.7)}$$

$$\underbrace{\mathbf{P(c \mid x)}}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}}\ \overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}} \quad \text{.................................(2.8)}$$

Posterior probability of class (c, target) given predictor (x, attributes) is denoted by P(c|x)

Prior probability of class is denoted by P(c).

Likelihood of the probability of predictor given class is denoted by P(x|c)

Prior probability of predictor is denoted by P(x).

The different types of Naïve Bayes Classifiers comprise multinomial Naïve Bayes, Bernouli Naïve Bayes and Gaussian Naïve Bayes.

2.3.5 Random Forest

Out of Naïve Bayes, Gradient boosting, Random Forest and decision tree [95], the maximum accuracy of 69.44% was obtained with random forest for classification of rice diseases. Maximum accuracy of 80.68% is achieved in [82] with the help of Random Forest and compared with other traditional machine learning classifiers.

2.3.6 Artificial Neural network

Artificial Neural Networks as in Fig 2.8 are unrefined electronic networks which are basically built over the neural architecture of brain. One record at a time is processed and the record's classification is matched with the investigated classification of the record. Errors are communicated back into the network for every record in order to alter the algorithm for upcoming iterations.

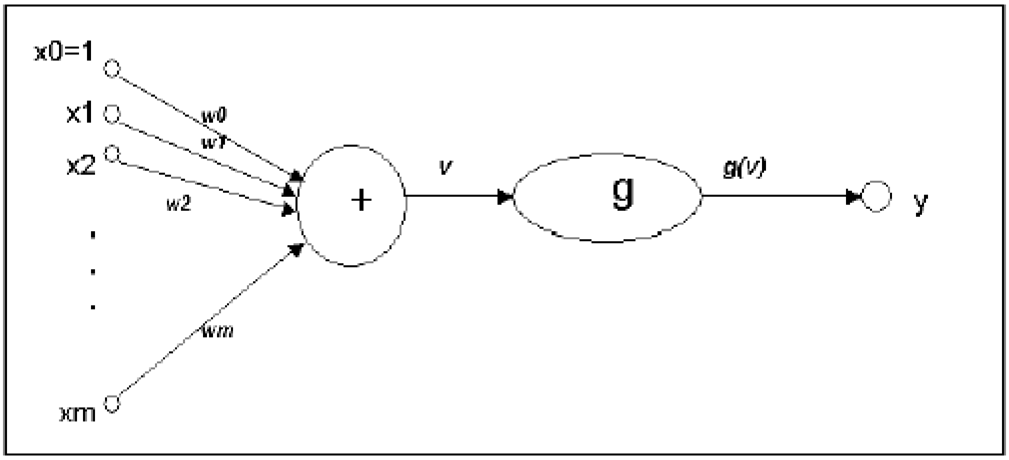


Fig 2.8 Artificial Neural Network

The input values are represented by xi and the related weights are represented by wi. A function (g) is responsible for summing up the weights and then plots the result to an output (y)

During the training period, the actual class of every record is known. The actual values can be allocated to the output nodes. For the actual class, 1 can be allocated while for the incorrect class, a 0 can be allocated. The learning process in ANN is iterative and therefore it is a significant feature of ANN. The records are communicated or fed to the network one after the other and the related weights with the input values are also modified. The same process is later on repeated. The adjustment of the weights helps the neural network in training itself for predicting the right class label for the given input values. Basically, it comprises three layers – Input layer, Hidden layer and the Output layer. [28] tested ANN along with the utilization of GLCM on two classes of cucumber disease - powdery mildew, downy mildew and healthy. Back Propagation Neural Network (BPNN) [41] is manifold layer feed forward network having three layer comprising 65536 neurons. In [69], BPANN has been utilized since SVM is considered to have less accuracy with regards to texture features. Also, GLCM has been utilized for texture features and for segmentation K-means has been used.

2.3.7 Convolutional Neural Network (CNN)

Convolutional neural networks (as in Fig 2.9) are a type of neural network that shares all of the properties of other neural networks. CNN was created expressly to interpret image data. The structure of their organization is therefore more selective: it is made up of two primary parts. The convolutional layer, the pooling layer, the ReLU correction layer, and the fully-connected layer are the four types of layers in a convolutional neural network.

- **The convolutional layer** – CNN’s significant element is the convolutional layer, which is always at least the first layer. Its goal is selection of characteristics in the photos that are given to it as inputs. This is accomplished by convolution filtering, in which a window depicting a characteristic on the picture is "dragged" and the convolution result between the characteristic and each piece of the digital image is calculated. In this case, a characteristic is viewed as a filter, and the two concepts are interchangeable. As a result, the convolutional layer takes multiple input images and computes the convolution of each with each filter. The filters match the features we're looking for in the photographs to a tree. We have a feature map for every combination (image, filter) that shows us where the features have been in the image: the greater the value, the more the associated location in the picture reflects the feature.

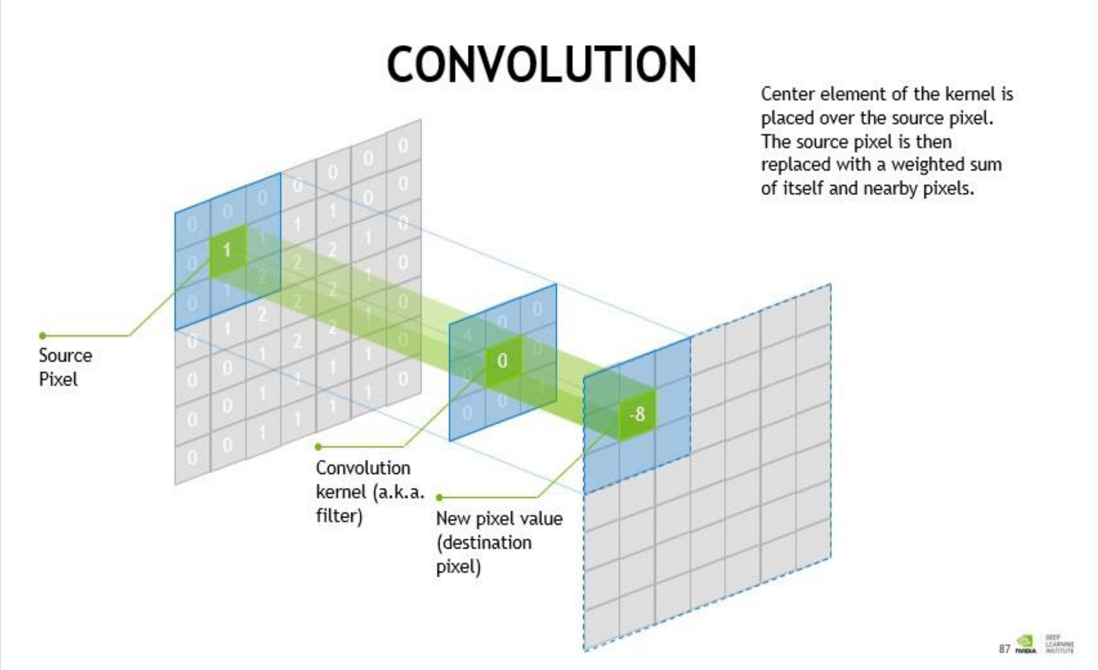


Fig 2.9 Convolutional layer [110]

- **The pooling layer** - This layer is frequently sandwiched among two convolutional layers. It accepts many feature maps and executes the pooling function to all of them. The pooling procedure diminishes the size of the photos while maintaining their essential properties. To accomplish this, we divide the image into regular cells and then preserve the highest value from each cell. Tiny rectangular cells are frequently utilized in practice to avoid missing too much data. The most popular options are 2x2 neighboring cells which don't crossover, or 3x3 cells isolated by a 2 pixel step (thus overlapping). The network's pooling layer minimizes the number of variables and computations. This increases network effectiveness and prevents over-learning. The highest values are recognized less precisely in the feature maps generated after pooling than in the ones obtained as input — this is a significant benefit. If you want to recognize a dog, for instance, you don't need to know exactly where its ears are: knowing that they're almost adjacent to the head is quite enough

- **The ReLU correction layer** - The real non-linear function defined by ReLU(x)=max is referred to as ReLU (Rectified Linear Units) (0,x). All negative values received as inputs are replaced by zeros by the ReLU correction layer. It serves as a mode of activation. It is shown in Fig 2.10.

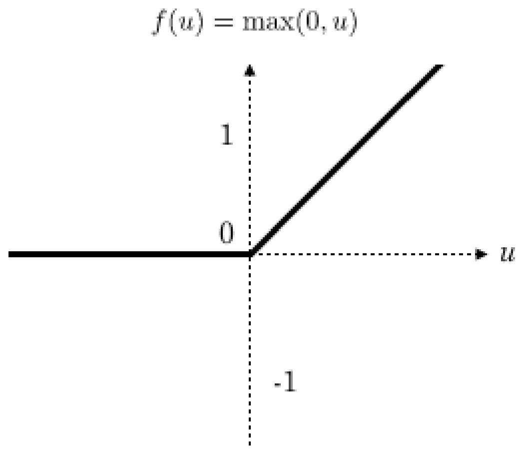


Fig 2.10 ReLU

- **The fully-connected layer** - The fully-connected layer, whether convolutional or not, is usually the last tier of a neural network, therefore it isn't unique to CNNs. This layer takes an input vector and turns it into a new output layer. It accomplishes this by applying a linear combination and, perhaps, an activation function to the incoming input values. The image is classified as an input to the system by the final completely connected layer, which produces a vector of size N, where N is the number of categories in the given image processing problem. Each member of the vector represents the likelihood that the supplied image belongs to a specific class. To determine the probabilities, the fully-connected layer computes product of each input element with weight, adds them together, and then performs an activation function (logistic if N=2, softmax if N>2) to each input element. This is the counterpart of multiplication of the input vector by the weights matrix. The term fully-connected points to the belief that every input values are linked to all target values. Weight values are learned in the same way as CNNs learn convolution layer filters: by back - propagation of gradient. The completely linked layer establishes the link between the image's feature positions and a class. Furthermore, the entry table correlates to a feature map for a certain feature, as it is the outcome of the preceding layer: the high values reflect the placement (more or less exact according to the pooling) of this feature in the picture.

2.3.8 Deep convolution neural network (DCNN)

For autonomous learning from training datasets, a CNN is used. CNNs can contain a lot of layers and neurons in them. DCNN (Deep Convolution Neural Networks) [50], [30] is a classification algorithm with four layers: a convolution layer with 20 filters, a 100-filter layer, a 1000-filter layer, and a 1500-filter layer. The SoftMax function makes use of four neurons in the output layers. 14 crop species have been identified using DCNN [31]. CNN [47] was used to classify ten rice illnesses.

2.3.9 Three Channel Convolution Neural Network (TCCNN)

Three color channels, three CNNs, fully connected, Softmax, and an output layer make up a unique type of CNN. TCCNN has been used to predict vegetable leaf disease [63]. TCCNN's channels receive one of these components. The convolutional features of each CNN are then collected and fed into the next convolutional layer and pooling layer. The collected features are then fused via a fully linked fusion layer, yielding a feature map that is deep level and used for identifying diseases. Ultimately, a softmax layer performs classification.

2.4 SURVEY ON MACHINE LEARNING TECHNIQUES

Haixia Qi et al. [85] adduced a unique approach by hybridizing the machine learning and deep learning techniques to detect the leaf diseases in peanuts. A total of 6029 images of diseased leaves of peanut plant have been scrutinized. 5 categories of diseases have been taken into account. The device used was a mobile phone to capture the pictures. Rotation, scaling and flipping are the three augmentation approaches adopted in this paper. Better accuracy has been obtained with deep learning model. The same deep learning model even performed better after it was used along with stacking ensemble and augmentation. The machine learning technique taken into account for stacking ensemble is logistic regression, RF and SVM. 97.59% of accuracy is achieved. ResNet50 and DenseNet121 have shown the maximum accuracy with logistic regression and RF with aid of data augmentation. The setup considered in the paper is a laboratory setup which is potential enough to obtain an accuracy of 97.59%. However, the natural environment is generally complex and hence detection of disease experiences complications. The authors have suggested a future work related to the same.

E. Samatha Sree Chaturved et al. [69] stressed that ability of humans to identify and analyze plant diseases is limited due to the fact that it is totally dependent on nanoscale activity. Computer-assisted picture rearrangement approaches are used in precise for plant disease classification and identification. On an obtained actual plant leaf image, a K-means clustering procedure has been used to determine disease. After detection has been completed, then we have a next step. The GLCM filter draws out features. The spatial frequency components, or how frequently a grouping of images illuminates contrast levels in pixels appearances, are bundled as GLCM. Feature extraction is the process of transforming contribution information into a group of spatial and texture statistic features. SVM based technologies are often used for classification. However, it has a low level of accuracy when it concerns to textural characteristics. To accomplish feature-formulated comparison, a BPANN technique based on innovative artificial intelligence is utilized for categorization. The proposed method has been tested in Matlab software, and its accuracy is far superior to that of standard methods. This analysis also includes applying a deep machine learning methodology to establish a generic disease classification for diverse infections.

Ms. Deepika Chauhan et al. [82] adduced a technique to project timely identification of diseases in maize crop. Almost all traditional classifiers have been compared on grounds of accuracy and F1 score. Feature extraction on grounds of color is performed. Appearance of the diseases leads to the variation in diseases. The color feature differentiates the diseases from one other. RGB feature extraction is conducted to filter out the color information from the images. 3823 images are observed for the experiment and 4 classes of diseases are taken into account. 90% of the images are utilized for training and the rest 10% for testing. Conventional classifiers such as

SVM, NB, KNN, DT and RF have been compared. RF is found to obtain the maximum accuracy of 80.68%. As a part of future work, the researchers of the paper have recommended the utilization of high-dimensional datasets.

Ahmed J. Afifi et al. [10] adduced a CBIR method with the aid of Ranklet Transform as a step in pre-processing and color feature. K-means clustering was utilized for this purpose. For making the image invariable towards rotation, Ranklet Transform has been utilized. As a part of future work, the authors have suggested the utilization of shape and texture features accompanying color features.

Yashwant Kurmi et al. [105] adduced a technique based on localization to categorize diseases in three crop types for distinct diseases. Initially, localization in the leaf zone is performed with the aid of color features and subsequently utilization of mixture model based region expansion. Different patterns are shown by the leaf images. These features of differentiating characteristics impact the classification of images. Characteristics of the infected leaves in the images such as spots and damaged area of the leaf in the localized pictures suggests the way the healthy images can be differentiated from the infected leaves and this can be easily done with the assistance of Fisher vector(FV) extraction. Gaussian distribution's differentiation of various orders aids the FV. FV is obtained by Fisher Kernel(FK). FK illustrates a feature by a gradient vector with the help of SIFT. The conduct of classification is determined on grounds of Area Under the Curve (AUC) and accuracy. The analysis is done on bell pepper, tomato and potato images of the PlantVillage dataset. Two categories of bell pepper, three categories of potato and ten categories of tomato have been considered. SVM and Multi-layer perceptron are utilized to test the performance. 94.35% is the maximum accuracy achieved and AUC is found to be 94.7%. In contrast with the existing techniques, the new approach is found to perform better. The authors have suggested the use of other crops as well as a future work to this research.

Sutha P. et al. [96] have put forward a fuzzy classification approach to recognize the diseases automatically in apple plant leaf. A fuzzy association function is applied that views the association between the pixels on grounds of the degree. 20 samples are stored in the database and captured using digital cameras. As a significant part of image processing, pre-processing step is also employed in the form of edge detection and thresholding. Otsu's approach is applied for thresholding the image with the utilization of histogram. Otsu's approach splits up the pixels into two categories – forepart and the backdrop. Various mathematical approaches are adopted for detecting edges. The edges are identified by taking into account the sharp variations in brightness of the image which ultimately aids in identifying the borders of objects in images. Color features and LBP features are extracted later on after employing the contrast enhancement technique in order to make the features in the image emerge prominently. Following this, K-means approach is utilized for segmentation purpose. Eventually, the fuzzy classification is employed to obtain the right class of disease. The adduced model offers an accuracy of 93% which is more in contrast to state-of-the-art techniques. The authors have suggested a future work as well. In future, modern background detachment approaches can be employed for splitting up the objects from the backdrop in the image. The adduced approach can also be merged with other approaches.

Vimal K. Shrivastava et al. [103] have presented a color feature based image based rice plant disease categorization technique. One of the most important characteristics used to distinguish rice plant infections is color. 14 distinct color spaces were investigated and four were extracted. There are 172 features in total, with features from each color channel. In this machine learning architecture, four statistical parameters were recovered and utilized as attributes. To classify rice plant diseased photos, the extracted color features were input to classifiers. Operating a 10-fold cross-validation process, the dataset was partitioned into training and test sets. Furthermore, the performance of seven different classifiers was evaluated, revealing that the SVM classifier achieved the greatest classification accuracy of 94.65%. There are four classes in the dataset. The dataset, which contains 619 photos, was used to train and test the models. The optimistic findings of this article reveal that color features can play an essential role in the development of a rice plant disease diagnostic model, allowing farmers to adopt proactive steps and improve product type and effectiveness. The authors hope to collect more datasets of rice plant illnesses with a larger number of labels in the future and use deep learning approaches to characterize rice plant diseases.

Sivagami S et al. [100] developed a new approach for diagnosing tomato plant illness. Pre-processing, segmentation, feature extraction, and classification are the four stages of the proposed technique. The obtained images were downsized and the noise was suppressed using the Weiner filtering technique in the pre-processing step. For that improved K-Means image segmentation algorithm was used. Segmentation was depicted as one of the crucial processes. Following the segmentation of the image, a Region of Interest (ROI) was identified, and relevant characters were collected using the feature extraction approach from this ROI. The authors employed the GLCM algorithm to extract significant factors from ROI in the suggested technique. Important characteristics are recovered from the segmented image utilizing the GLCM feature extraction approach after the segmentation stage. Eventually, classification techniques such as Support Vector Machine (SVM) and Adaptive Neuro Fuzzy Inference System (ANFIS) are used to categorize sick leaves. The tests are based on photos of tomato leaves found in PlantVillage database. The proposed scheme is put to test in tomato plants for five different illnesses. The adduced advanced K-Means with ANFIS classification algorithm achieved the best accuracy of 98.60%.

Jaweria Kianat et al. [86] suggested a hybrid paradigm built on fusion of feature and selection algorithms that uses three fundamental phases to classify cucumber illnesses. The contrast of picture samples is improved in the first stage, which is followed by feature extraction, fusion, and selection in the second stage, which is started with augmenting the data. Eventually, a collection of classifiers is employed to classify the majority of distinguishing features. The proposed probability distribution-based Entropy (PDbE) methodology is used to decrease the retrieved features in this study. Powerful features are picked using the suggested Manhattan distance controlled entropy (MDcE) after the serial based fusion stage. It is noticeable that the offered method is analogous to several other current strategies based on the attained accuracy (93.50%) on the selected dataset of over 900 sample images and six categories.

Appasaheb Gargade [80] suggested approach to identify illness in Custard Apple disease. A framework for crop parameter analysis, identification of N, P, K deficits, and leaf illnesses is offered. The algorithms KNN and SVM are employed to assess leaf deficits and illnesses. For leaf illnesses and deficits, a library of 125 and 80 Custard apple leaf photos is employed. The offered leaf parameter measurement technique was found to be 99.5% accurate in experimentation.

Sukhvir Kaur et al. [52] developed rule based semi-automated framework based on K-means principles that was created and executed to differentiate normal leaves from infected leaves. Furthermore, a diseased leaf is categorized into one of three groups. Tests are carried out by training three models based on SVM using color and texture characteristics, and their blends individually. Thousands of pictures from PlantVillage's repository were used to get the outcomes. All of the studied mixtures have satisfactory average accuracy values that are also shown to be superior to previous techniques. The goal of this research was to find the optimal characteristic set for monitoring leaf disorders in soybeans. Using a large-scale dataset of 4775 photos, the average maximum classification accuracy observed was 90%. Visual analysis of the test photos further validates the suggested system's approval. In the long term, the investigation might comprise real-time visuals for both assessment and training. The study can indeed be expanded to include 3-D leaf images. After noticing several segmentation properties for the training pictures, the rules are constructed. As a result, in the not-too-distant future, an endeavour to construct a fully autonomous mechanism for both recognition and categorization can be made. Moreover, the present investigation excludes some disease classes; nevertheless, this scope can also be expanded.

Anupama S. Deshapande et al. [55] devised a method for detecting common fungi in maize leaves. Using first-order histogram attributes and Haar wavelet characteristics based on GLCM, the developed scheme seeks to detect diseases early and then classify them as diseased or healthy. The study takes into account two classifiers: KNN and SVM. For K = 5, KNN has the maximum efficiency of 85%, whereas SVM based classification has a maximum efficiency of 88%. With minimal changes, the work can be broadened to uncover other maize plant disease as well as other plant diseases.

Shanwen Zhang et al. [53] presented an IOT based segmentation and identification approach in accordance with blending of Super-pixel and K-mean clustering, and pyramid of histograms of orientation gradients (PHOG) methods. To begin, a super-pixel grouping algorithm converts the colored sick leaf image into a few packed super-pixels. The diseased picture is then split from each super-pixel using the K-means clustering technique. Subsequently, the PHOG characteristics are derived from three color elements of each separated diseased picture and its grayscale picture, and four PHOG descriptors are compiled as a vector. The proposed strategy appears to be effective based on the outcomes of two plant sick leaf picture datasets. This research proposes a practical method for segmenting and recognizing plant sick leaf pictures.

Vijai Singh et al. [45] offer a strategy for picture segmentation that may be used to detect and categorize plant leaf ailments automatically. It also includes an overview of various disease categorization systems that can be used to detect plant leaf ailment. The genetic algorithm is used to do image segmentation, which is a crucial component of disease detection in plant leaf disease. The suggested algorithm's overall classification accuracy is 97.6%.

Shanwen Zhang et al. [34] present a cucumber disease recognition approach based on the Gglobal-Local Singular Value Decomposition (SVD) to improve the classification performance of cucumber disease. The spotted picture is first split from every cucumber infection plant leaf using the watershed technique. Secondly, every spotted photograph is separated into a few blocks, and the SVD algorithm retrieves and orders the merged features of Global-Local singular values out of each block. Thirdly, the key-point vectors are built and their dimensionalities are modified to be equal. Finally, an SVM classifier is used to identify the unknown illness leaf image's class. When compared to current cucumber disease recognition algorithms, the recommended one can retrieve key-point characteristics from a spotted picture with a much smaller dimension than the actual space. Three types of cucumber disease leaf photos are used to test the method. The results of the experiments reveal that the adduced method is useful and viable for detecting cucumber disease, and that it has the maximum classification accuracy and practical usefulness. The fundamental flaw in the method provided in this paper is that extracting the singular values of a few sub-blocks demands more computing time. More work is needed to completely use the proposed technique in large-scale image-processing systems, with the vast dataset and sophisticated color extraction characteristic containing higher recognition results being the primary issues.

Alexander Johannes et al. [36] describe a unique image processing approach on the basis of candidate hot-spot detection in conjunction with statistical inference approaches to combat illness diagnosis in wild situations. The reliability of early detection of three European chronic wheat illnesses is examined in this study. The research was carried out with the use of seven mobile devices and over 3500 photos collected at two test sites in Spain and Germany. On the pilot tests in real-world situations, AUC metrics more than 0.80 were found for all of the examined disorders.

Min Zhang et al. [9] offer a new method based on global characteristics and zone based local characteristics for detecting citrus canker using leaf images taken in the outdoors, which is more problematic as compared to leaf photographs taken in labs. To begin, the most important properties of citrus lesions are selected using an updated AdaBoost algorithm for segmentation of the disease from their backdrop. Subsequently, a canker lesion identifier is presented, which incorporates both the color and local texture organization of canker infection zones as indicated by plant phytopathologists. To recognize canker lesions, a two-level hierarchical detection system was created. Finally, the suggested method is tested and compared to previous ways, with the current outcomes indicating that the suggested scheme yields classification accuracy comparable to that of domain analysts. Future research will imitate experts' observations of how to merge multi-angle photos of a citrus leaf for recognition and apply the proposed method to disease monitoring and quality control in several other crops.

J. G. A. Barbedo [27] presented an algorithm for distinguishing plant disease signs and symptoms from symptomless tissues in plant leaves. The H (from the HSV color space) and a (from the L*a*b* color space) color channels' histograms are modulated by this simple technique. The suggested method was put to the test in a range of scenarios, including 19 species of plants, 82 illnesses, and photos collected in both controlled and unregulated environments. The method has proven to be useful for a wide range of plant illnesses and ailments, while some cases may necessitate other approaches.

Xuebing Bai et al. [46] proposed an advanced FCM methodology. Analysis was carried on 129 cucumber infection photos in the vegetable illness dataset to assess the accuracy and efficiency of the suggested segmentation algorithm. The average segment error is just 0.12%, according to the findings. The new framework for categorizing and rating apples in cucumber disease diagnosis is successful and resilient, and it can be simply extended for additional spectroscopy agricultural functions.

Sourabh Shrivastava et al. [19] describe a completely automated disease diagnostic and level estimate system based on color picture sensing and computation. Several novel factors, including Disease-Severity-Index (DSI), Infection-Per Region (IPR), and Disease-Level-Parameter (DLP), for evaluating disease severity and level-classification have indeed been developed. An actual library of Soya leaves gathered between July and September 2012 was used to evaluate the suggested approach. Experiments have proven that the technique outperforms the alternatives. The platform's performance can be enhanced in the near future by extracting the leaf item from a complicated backdrop using sophisticated scene removal strategies.

Khushal Khairnar et al. [72] proposed an approach for cotton plant leaf disease diagnosis. The introduced method is built on image processing techniques, and it begins by segmenting the damaged cotton plant leaf picture using the K-means algorithm. The separated image is then used to derive color and texture attributes. SVM is used to diagnose diseases based on feature segmentation.

Devashish Pujari et al. [26] used feature reduction method to identify plant diseases based on SVM and ANN. In a need to experiment with the sample photographs of plant diseases, color and texture attributes were utilized. Color and texture characteristics are extracted using algorithms that are then used to train SVM and ANN classifiers. A restricted feature set based strategy for recognizing and classifying photos of plant diseases was described in the study. Using only color features with SVM, the average identification accuracy is 84% while using only texture features with SVM delivered 89% accuracy. Using only color features with ANN, the average identification accuracy is 82% while using only texture features with SVM delivered 84% accuracy. The results show that the SVM classifier is better for identifying and classifying plant disorders that alter agriculture and floriculture crops. Accuracy of 92.17% was found with SVM.

Radhakrishnan Sreevallabhadev [77] proposed machine learning algorithm to predict blast disease in paddy plant. Pre-processed photos of diseased and normal rice crops are fed into machine learning approaches using CNN for extracting the features and SVM is used to categorize in the proposed system. For both contaminated and normal photos of paddy blast illness, the results reveal that CNN paired with an SVM based classification technique delivers greater accuracy than SVM alone. Producers would profit greatly from the proposed strategy in terms of higher revenue, and the community as a whole would profit from food safety. The similar machine learning technology could be used to diagnosis other rice crop illnesses in the future.

Gittaly Dhingra et al. [56] employ an innovative fuzzy set derived from neutrosophic rule based segmentation method to assess the zone of interest in this study. Three membership elements identify the divided neutrosophic image: true, false, and intermediate region. Major feature subsets based on segmentation results, such as texture, color, distribution, and infection pattern region, are examined to determine if a leaf is diseased or healthy. In addition, nine distinct classifiers are utilized to track and illustrate the discriminatory strength of blended feature efficacy, with random forest outperforming the others. A total of 400 cases (200 normal and 200 unhealthy) were used to evaluate the suggested system. The proposed methodology might be a useful tool for identifying diseases in leaves. A new set of features appears promising, with a recognition rate of 98.4%. Future research could target on expanding on the current work by classifying each illness group separately and estimating the intensity of the illnesses discovered. To improve the performance of illness detection and diagnosis models, an unexplored combination of extracting features, feature selection, and learning strategies can even be investigated.

Raihan Kabir et al. [75] proposed an effective leaf disease identification model that uses superior segmentation methods to derive discriminant characteristics from disease split leaf images. Employing image segmentation, the diseased and healthy sections of the leaf are differentiated in this work. Then, based on the color and luminance values of the pixel values of the leaf image, several distinguishing characteristics are retrieved. Ultimately, the leaf illness is detected and diagnosed using MC-SVM with RBF Gaussian kernel. A benchmark dataset PlantVillage Dataset is utilized to verify the suggested model. The proposed model's illness detection ability was evaluated and found to be acceptable. In addition, three excellently known image segmentation methods are implemented, and the best segmentation approach is examined in order to improve disease diagnostic accuracy. According to the results of the trial, the Otsu thresholding based ailment segmentation produces the best results for leaf diagnostics.

Md. Tarek Habib at al. [73] offer an online computer vision based agro-medical intelligent system that analyzes an image acquired with a smartphone or portable device and diagnoses ailments in order to assist far-flung farmers in resolving the issue. To demonstrate the effectiveness of the suggested intelligent system, several experimental tests were carried out. To begin, they offered a set of qualities from the perspective of differentiating characteristics. The K-means segmentation approach is utilized to split out the disease affected zone from the collected image, and thereafter essential characteristics are retrieved using a SVM classifier to identify the disorders. When compared to recent publications, a recognition rate of more than 90% has been reached, which looks good as well as encouraging. Future study with a big data collection of photos to encompass a broader spectrum of papaya illnesses is still possible.

H. Ali et al. [39] described a method for detecting and classifying serious citrus diseases that are economically significant. The kinnow mandarin, which accounts for 80% of Pakistan's citrus industry, was the survey's main emphasis. The method utilizes a Differential Evolution (DE) color difference algorithm to distinguish the disease affected region, as well as a color histogram and textural properties to identify diseases. The approach outperformed the competition, achieving total accuracy of 99.9% and sensitivity of 0.99 AUC. Furthermore, trials were conducted using a mixture of color and texture, which produced similar results when compared to separate channels. The dimensions of the features were minimized using Principle Component Analysis (PCA), and the minimized features were then assessed using state-of-the-art classifiers. More training examples can boost the accuracy of the suggested method's outcomes. Standardized publicly accessible datasets are still required to enhance the efficiency of such platforms and to expand the scope of computer assisted assessment systems capable of reliably identifying and classifying various disorders.

Kuldeep Singh et al. [59] wanted to locate and recognize the initial signs of rust illness at the molecular scale. On microscopic images, the performance of different pre-treatment, extraction of features, and classification algorithms was tested. Eventually, an SVM classifier was employed to detect Pea Plant leaf infection. With an accuracy of 89.60%, the proposed method may properly recognize and evaluate diseases. Early diagnosis of leaf rust at the molecular level has been emphasised in order to prevent disease transmission not only on the complete plant but also to neighbouring crops.

X.E. Pantazi et al. [64] proposed a method that uses LBPs for extracting features and One Class Classification for categorization to provide an automated method of crop disease diagnosis on multiple leaf sample images matching to diverse crop plants. For every plant medical condition, the developed system employs a specialised One Class Classifier. The techniques that were trained on grape leaves have been examined in a range of crops and have shown to have a high level of generalization when applied to other species. When ambiguous data samples may correspond to one or more categories, an original technique providing dispute resolution between One Class Classifiers gives the proper categorization. For the 46 plant disease combinations examined, a total rate of success of 95% was reached.

Wajahat Kazmi et al. [22] looked at how to detect thistles in sugar beet farms in realistic, outdoor settings. They used a standard color camera in their research and derived vegetation indices from the photographs. A total of 474 sugar beet and thistle field pictures were gathered and sorted into six categories based on brightness, height, and age. There were 14 indices in the set of features. After a PCA analysis, the set of features was limited to four significant indices, but the recognition rate was comparable to that achieved by combining ExG and GB, which was about 95%, substantially superior to an individual index. Multistep linear regression chose nine of the 14 features with a 97% accuracy rate. Linear Discriminant Analysis (LDA) and Mahalanobis Distance (MD) had similar results, leaving them both evenly acceptable. Ultimately, the findings were confirmed by employing the learned classifiers to annotate photos including both sugar beet and thistles. The testing trials revealed that the two main essential factors impacting the identification are sunshine and the size of crop, which would be connected to its growth phase. Photos of immature sugar beet (in the seventh week) in a shelter produced the greatest outcomes in this research. In the future, the authors want to use the morphologies of the leaflets' edge segments in conjunction with color indices to recognize weeds instead of the high computational job of segmenting the leaf.

Malusi Sibiya et al. [61] discuss a proposed technique for estimating the degree of leaf diseases employing maize leaf damaged specimen. A number of scholars have investigated the subject of estimating the degree of leaf diseases in the field, but only a few have employed fuzzy logic to estimate severity predictions of leaf diseases. The goal of this study is to introduce the advantages of fuzzy logic decision making criteria to the proposed method used in the "Leaf Doctor" application, which is used to evaluate the severity of leaf diseases. This strategy will help advance precision farming techniques by introducing a method that can be implemented in mobile phones and used in apps like the "Leaf Doctor." The programs created using the technique suggested in this work will assist users who are not plant scientists in determining the severity of the predicted infection. Use of fuzzy inference system with image segmentation distinguishes it from other methodologies.

2.5 SURVEY ON DEEP LEARNING TECHNIQUES

Sakshi Mangal et al. [97] used image processing method such as the Laplacian filter and Unsharp masking approach, as well as image segmentation strategies such as Canny edge detection, to detect early disease in Pepper bell, Potato and Tomato plant. The classification model used in this research is a deep learning categorization model called convolution neural network. Kernel sharpens the image using the Laplacian filter. Kernel is a 3*3 matrix that is twisted with the image matrix to obtain the required sharpening effect. The authors used canny edge detection methodology to identify and segment the entire image based on edges. The GLCM method is used to extract characteristics from leaves. The GLCM is a tool for detecting texture and spatial features in photographs. All features are captured in the convolution layer by creating a feature map, which is created by convolving an image matrix with a 3*3 filter matrix. The authors used a confusion matrix to determine the model's performance. In the confusion matrix, we first locate true positives, in which we provide correct data and the model accurately predicts it. Second, we uncover true negatives, which occur when we provide valid data but the model incorrectly predicts it. Thirdly, we have false positives, which occur when we submit incorrect data yet the model predicts a proper result. Fourth, if we submit inaccurate data and the model likewise predicts that it is erroneous, we get a false negative. With 97.82% accuracy and an average validation loss of 10%, the authors developed a convolution neural network.

Jaemyung Shin et al. [87] employed Deep Learning (DL) to discover powdery mildew, a severe fungal infection in strawberries to limit the amount of unneeded fungicide use and the requirement for field scouts. AlexNet, SqueezeNet, GoogLeNet, ResNet-50, SqueezeNet-MOD1, and SqueezeNet-MOD2 were among the well-known learners optimized and evaluated in this work. To avoid overfitting and to account for the varied shapes and directions of the leaves in the field, data augmentation was undertaken on 1450 healthy and sick leaf pictures. Overall, the six DL approaches employed in this work had classification accuracy greater than 92% on average.

M.Yogeshwari et al. [88] applied GLCM characteristics and convolutional neural networks to assess plant leaf sickness. Using GLCM, characteristics are recovered from segmented pictures. PCA is utilized to lessen the dimensionality of features. Finally, a unique DCNN architecture is employed for categorization. Four convolutional layers, two fully linked layers, and one SoftMax layer make up the suggested architecture. The developed model is reliable and delivers the highest classification results when compared to existing classifiers, according to empirical observations.

Juncheng Ma et al. [50] suggested using a deep convolutional neural network (DCNN) to recognise four cucumber diseases based on their symptoms. Cucumber leaf pictures obtained in the field were subdivided to create the symptom photos. Few images were collected from PlantVillage dataset. Data enlargement methods were used to extend the datasets created by the segmented symptom photos in order to reduce the risk of overfitting. The DCNN showed good recognition performance with the supplemented datasets containing 14,208 symptom photos, with an accuracy of 93.4%. Comparative tests were performed utilizing traditional classifiers such as RF, SVM and AlexNet, to compare the DCNN outcomes. The DCNN proved to be a reliable method for detecting cucumber infections in arid environments.

Guan Wang et al. [38] developed a series of DCNNs that are trained to assess the severity of the disease using the apple black rot photos in the PlantVillage dataset, which are further described by biologists with four severity levels as ground truth. In this research, the systemic performance of shallow networks generated from scratch and deep models fine-tuned via transfer learning is examined. On the hold-out test set, the best system is the deep VGG16 model built with transfer learning, which has a best average of 90.4%. The proposed deep learning approach has a huge amount of potential for modern agriculture disease prevention. Additional data at various phases of different diseases will be obtained in the future using adaptable sensors such as infrared and hyperspectral cameras. Treatment suggestion, yield prediction, and other applications can all be linked to the deep learning algorithm.

Konstantinos P. Ferentinos [51] used deep learning approaches. CNN models were constructed to detect and evaluate plant diseases using simple leaf images of healthy and diseased plants. The models were trained using an open collection of 87,848 photos, which included 25 different plants in 58 different classes of [plant, illness] pairs, comprising non infected plants. Numerous model architectures were given training, with the top performing one achieving a success rate of 99.53% in detecting the corresponding [plant, illness] pair. The model's critical success factor makes it a valuable advising or early warning tool, and it's a method that might be broadened to accommodate an integrated plant disease diagnosis tool that can function in practical applications.

Srdjan Sladojevic et al. [33] is interested in using DCNNs to establish a new way to developing a crop diseases recognizer based on leaf image categorization. The methodology employed and the novel technique of training allow for a quick and painless system setup in practise. With the capacity to spot plant leaves from their surroundings, the built model can differentiate 13 different types of plant illnesses from healthy leaves. The deep CNN training was carried out using Caffe, a deep learning architecture. For independent class examinations, the experimental findings on the constructed model attained precision around 91% and 98%, and on average 96.3%. In addition, future study will include broadening the model's application by training it to diagnose plant diseases across larger geographical regions, merging drone-captured satellite images of fruit orchards with CNNs for object tracking.

Juanhua Zhu et al. [70] suggested an autonomous detection technique for grape leaf diseases based on image analysis and a BPNN. The illness images were processed using the Wiener filtering approach built on the wavelet transform. The Otsu approach was utilized to segment the grape leaf disease zones, and geometrical algorithms were intended to enhance the disease appearance. To capture the entire edge of the lesion site, the Prewitt operator was used. Radius, area, roundness, rectangularity, and geometry complexity were all recovered as significant dynamic characteristics. The recommended detection method for grape plant disease might be utilized to evaluate grape illnesses with significant accuracy rate, according to the findings.

Keke Zhang et al. [60] used CNN to uncover Xanthomonas campestris-infected peach leaf disease. AlexNet was fine-tuned using transfer learning. The power of self-learned properties is demonstrated by feature visualization from the trained CNN model. In order to evaluate the performance of CNN with standard classification methods such as SVM, KNN, and BPNN in detecting peach leaves, three comparative tests were run. The confusion matrix for each result revealed that CNN can identify Xanthomonas campestris affected peach leaves with 100% accuracy. With an AUC value of 0.9999, Receiver Operating Characteristic (ROC) curves and AUC values, an average performance evaluation, demonstrate that CNN performs better. The notable observation indicated that CNN is massively better than the other three approaches, with p-values of 0.0343 (vs.SVM), 0.0181 (vs.KNN), and 0.0292 (vs.KNN) respectively (vs.BP). In a nutshell, CNN outperforms the current techniques in detecting damaged peach leaves.

Geetharamani G. et al. [57] presented a Deep CNN based method for identifying plant leaf diseases. An available collection with 39 multiple categories of plant leaves and background photos is used to build the Deep Neural network. Using leaf pictures, the suggested Deep CNN model can accurately categorize 38 different types of healthy and ill plants. Furthermore, the data augmentation boosts the number of training data points from 49,598 to 55,636. An expanded dataset of 61,486 images and 30 0 0 training epochs was used to train and validate the most proficient Deep CNN model. In the categorization of the experimenting set plant leaf photos, the suggested model obtains an average accuracy of 96.46%, with individual category accuracy ranging from 92%to 100%. In order to expand the variety of dataset categories and the database's volume, this project will collect fresh photos from a variety of sources of diverse plant species, geographical locations, foliage growths, cultivation settings, picture resolutions, and types. Using various fine-tuning approaches, the expanded database will increase the model's accuracy and performance.

Yang Lu et al. [47] suggested a unique rice disease identification approach built on DCNNs techniques. CNNs are trained to recognise 10 common rice diseases using a collection of 500 real photos of damaged and normal rice leaves and stems obtained from a rice experimental site. The suggested CNNs based approach obtained a

95.48% accuracy using a 10-fold cross-validation technique. This accuracy is far superior to that of a traditional machine learning algorithms. The recommended method's practicality and effectiveness are demonstrated by numerical simulations for the monitoring of rice illnesses. The authors intend to use alternative deep models and training strategies in future research, such as the limited Boltzmann machine, which performs better on object identification. This model can be used to diagnose faults. In addition, the authors intend to conduct a more thorough examination of the training approach using both labelled and unlabelled specimens. The findings of this research could be used to sense devices and nonlinear time-dependent systems with distributed predictive control difficulties.

Mostafa Mehdipour Ghazi et al. [43] employ DCNNs to recognize the types of plants taken in a photograph and assess the effects of various conditions on the networks' efficacy. This is accomplished using three powerful and well-known deep learning architectures: GoogLeNet, AlexNet, and VGGNet. Using the LifeCLEF 2015 crop task databases, transfer learning is applied to calibrate the pre-trained systems. Data augmentation approaches are used to reduce the risk of overfitting. To boost overall performance, the networks' parameters are modified and numerous classifiers are blended. On the testing set, the highest combined algorithm had an accuracy rate of 80% and an average inverse rank value of 0.752 on the formal test set. When compared to the results of the LifeCLEF 2015 leaf recognition operation, the authors have increased the top system's overall prediction performance by 15% and its overall inverse rank score on the test set by 0.1, while surpassing the top three challenge players in all aspects.

Aditya Khamparia et al. [66] present a novel method for detecting plant disease using crop leaf pictures and convolutional encoder networks. The results were achieved using a 900 picture collection, with 600 images serving as the training set and 300 serving as the test set. There are three crops and five types of crop diseases to consider. The suggested network has been trained to recognize plant diseases from leaf photos. In the suggested work, different convolution masks such as 2x2 and 3x3 are applied. It was discovered that the proposed framework yielded varied levels of accuracy depending on the number of time periods and the size of the convolution layer. In 100 epochs, the researchers achieved 97.50% accuracy for a 2x2 convolution kernel size and 100% accuracy for a 3x3 convolution kernel size, which is superior as compared to other standard approaches. This study can also be expanded to address a variety of common plant diseases, which may be more advantageous in farming to increase output. There are various more hyper-parameters that can be used to calibrate the suggested model, such as dropout and normalisation.

Karthik R et al. [76] introduced two distinct deep models for diagnosing the class of infectious disease in tomato leaves. The first architecture uses residual learning to learn important categorization characteristics. On the basis of the residual deep network, the second architecture employs an attention method. PlantVillage Dataset was used in the investigations, which included three diseases: early blight, late blight, and leaf mould. By utilizing attention method, the suggested work leveraged the features learnt by the CNN at different production hierarchies and attained a 5-fold cross-validation rate of 98% on the validation datasets.

Yusuke Kawasaki et al. [21] offer a unique CNN based leaf disease identification system. CNN can autonomously collect the required features for classification and accomplish good classification accuracy using only training photos. This novel methodology was utilized to train CNN using 800 cucumber leaf pictures. The suggested CNN based solution gives an average performance of 94.9% in distinguishing cucumbers into two usual illness classes and a non-diseased class utilizing a 4-fold cross-validation technique.

Ümit Atila et al. [102] stressed that there is a requirement for a model which does not demand pre-processing and can conduct an effective categorization rather than conventional machine learning methods, wherein the mechanical extraction of features must be faultless to achieve efficient outcomes. In this paper, the EfficientNet deep learning network was suggested for plant leaf disease classification, and its performance was compared to that of other state-of-the-art deep learning architectures. Models were trained using the PlantVillage dataset. All of the models were given original and unique training. 55,448 and 61,486 photos were added to the supplemented datasets, respectively. Transfer learning was used to train the

EfficientNet architecture and other deep learning techniques. All layers of the models used in transfer learning were created with the intention of being trainable. In the test dataset, the B5 and B4 models of the EfficientNet architecture achieved the greatest accuracy and precision scores when compared to other deep learning models, with 99.91% and 99.97% for accuracy and 98.42% and 99.39% for precision, respectively. The plant leaf disease dataset will be broadened in the future by enhancing crop variety and the number of training. This will facilitate the development of models that could make more correct estimates in challenging circumstances. Plant physicians and growers will be able to efficiently make a diagnosis of plant diseases and exercise caution if these enhanced models are configured in mobile environments.

Y. A. Nanehkaran et al. [79] presented a unique approach for detecting plant leaf infections. Segmentation process and image classification are the two aspects of the procedure. For the disease sign segmentation of plant disease photos, a hue, saturation, and intensity based and LAB based composite segmentation method is suggested and deployed first. The segmented photos are then fed into an image categorization CNNs. The recognition rate attained using this method was about 15.51% greater than that obtained using the traditional method. Furthermore, the detection findings revealed that under difficult background settings, the average success rate was 75.59%, and the majority of disorders were efficiently recognized.

Sanjay Patidar et al. [78] provide a paradigm for detecting and classifying infections in rice plants, one of the most important crops in the Indian staple food. The focus was mostly on three disorders. The UCI Machine Learning Repository's Rice Leaf Disease Dataset was used. To categorize the photos into the appropriate disease groups, a Residual Neural Network was utilized, which has been discovered to be a quick, efficient methodology that outperforms basic CNN and other classifiers like the SVM by preventing the system from reaching fullness for larger data or bigger networks. On the dataset, an accuracy of 95.83% is reached.

G. Sambasivam et al. [84] used 10,000 tagged photos taken during a routine investigation in Uganda to detect cassava diseases using 5 fine-grained cassava leaf disease types. As a result, the studies concentrated on strategies for building deep

CNNs from foundation with class weight, Synthetic Minority Over-sampling Technique (SMOTE), and focal loss to attain an overall accuracy of over 93%. The intention was to correct for high class inequality so that the algorithm could reliably forecast marginalised categories.

Jing Chen et al. [58] created a deep CNN to distinguish tea crop diseases types from leaf photos. LeafNet is a CNN’s architecture with varied sized feature extractor layers that collects the characteristics of tea plant illnesses from photos automatically. Dense Scale Invariant Feature Transform (DSIFT) characteristics are also retrieved and utilized to build a Bag of Visual Words (BoVW) model, which is then used with SVM and MLP classifiers to identify illnesses. The three classifiers' disease recognition performance was then assessed independently. The LeafNet architecture correctly classified tea leaf diseases the most, with an average accuracy rate of 90.16%, compared to 60.62% for the SVM method and 70.77% for the MLP method. When compared to the MLP and SVM algorithms, the LeafNet algorithm was highly comparable in recognizing tea leaf illnesses. As a result, the LeafNet can be utilized to enhance the productivity and reliability of disease diagnosis in tea plants in the long term.

Bin Liu et al. [68] suggests an innovative identification strategy based on upgraded CNNs for the assessment of grape leaf diseases. First, using image pre-processing techniques, a data collection of 107,366 grape leaf images is constructed based on 4,023 photos obtained from field and 3,646 photos acquired from publicly available data sets. Following that, the Inception framework is used to improve the performance of multidimensional extraction of characteristics. In addition, to increase reuse of feature and strengthen feature dissemination, a layered connectivity technique is adopted. Finally, Dense Inception Convolutional Neural network (DICNN), a unique CNN based model, is designed and developed from the ground up. Under the hold-out test set, it achieves an accuracy rate of 97.22%. The accuracy performance improves by 2.97% and 2.55%, respectively, as contrasted to GoogLeNet and ResNet-34. The current outcomes show that the suggested model is capable of accurately detecting grape leaf illnesses. This research investigates a novel methodology for the fast and effective identification of plant diseases, laying the theoretical groundwork for the use of deep neural networks in agricultural data.

Ankur Das et al. [65] built an automatic feature engineering framework for rice leaf disease identification on the basis of deep learning. The goal of this project is to improve crop productivity by forecasting leaf diseases proactively and adopting actions to eliminate or at least stop disease transmission in the neighbouring regions of diseased areas. The damaged section of the rice plant's leaves is first detected and isolated from the rest of the leaflets. The CNN model, a current family of deep learning methods, is fed with these sick leaf images. Four layers of convolution, two fully connected layers, and a softmax layer of output make up the CNN model's structure. The key reason for utilizing CNN is that it has the ability to construct an endless range of attributes with no inherent influence, thanks to automated feature engineering. It can also capture all complicated nonlinear connections between attributes. The disorders are then categorized using several classifiers after a dimensionality reduction strategy is used to eliminate redundant attributes. A total of 10,500 contaminated leaves were used to test the procedure. The test findings and comparative analysis regarding performance assessment demonstrate the efficacy of the suggested strategy and aids in the selection of best classifier diagnose rice leaf disorders.

2.6 COMPARISON CHART

Table 2.2 shows the comparison chart for literature survey

Table 2.2 Comparison chart for literature survey

Paper	**Year of Publication**	**Technique**	**Main findings**
[9]	2011	Local Binary Pattern on Hue, Color and texture features, AdaBoost algorithm	87.99% accuracy
[14]	2015	Fuzzy C-means clustering	88% accuracy
[34]	2016	Global-Local SVD(GL-SVD),	91.63% recognition

		SVM	rate
[53]	2017	K-means, PHOG, SVM	Apple – 85.64% Cucumber – 87.55%
[36]	2017	Global color constancy	Average accuracy of 78.166%
[59]	2018	Discrete Wavelet Transform, SVM	Accuracy of 89.60%
[73]	2018	Co-occurrence Matrix Features, Statistical Features, K-means, SVM	90.15% accuracy
[52]	2018	Color, GLCM, Gabor, 2DWT	Approx. 90% accuracy
[64]	2018	LBP	Success rate of 95%
[55]	2019	Haar wavelet features based on GLCM features	Highest accuracy of 88% obtained with SVM
[62]	2019	Color based feature extraction	SVM classifier with SOS as the optimization algorithm and was found to be 90%.
[103]	2020	Color Feature Extraction, SVM, DC, KNN, NB, DT, RF	94.68% accuracy
[105]	2021	Color features, Naive Bayes (NB), Decision Tree (DT), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest (RF).	Highest accuracy of 85% recorded in the RF classifier
[106]	2021	Stacking ensemble	97.59% accuracy

2.7 RESEARCH GAPS

- Convolution based Law's mask features [17][29][42] are not utilized in state-of-the art techniques. Convolution based features reduce the noise in features and therefore, help in calculating the statistical features efficiently.
- Segmentation is not optimized [16] in state-of-the art techniques, which actually increases noise in features and reduce accuracy. Segmentation needs to be optimized using an efficient and latest meta-heuristic approach which helps in achieving efficient classification results.
- Ensemble classification of machine learning classifiers has never been tested in the state-of-the-art techniques. An ensemble of models is a collection of learning models whose individual predictions are integrated so that constituent models mitigate for each other's flaws. This will improve the classification results.

2.8 DATASET

We have used the PlantVillage dataset that contains 54,306 photos of plant leaves with 38 different class designations. A total of five crop types from the mentioned dataset – Bell pepper, Potato and Tomato have been considered. For bell pepper, bacterial spot and healthy classes are considered. For potato, early blight, late blight and healthy classes are considered. For tomato, bacterial spot, early blight, healthy, late blight, leaf mold, septoria leaf spot, spider mites, target spot, mosaic virus, yellow leaf curl are considered. Table 2.3 presents the categories of plant, the corresponding diseases used in our work and the state-of-the-art techniques used for each plant type.

2.8.1 Bell pepper dataset

The PlantVillage dataset contains the plant image dataset library for image based disease analysis. There are 54,309 tagged photos in 14 distinct cropping dataset. There are two types of bell peppers in the database: healthy and disease. Fig 3 shows a selection of photographs from the database. The dataset has 997 bacterial spot photos and 1478 healthy class images.

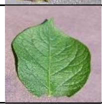
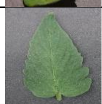
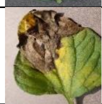
2.8.2 Potato dataset

Potato images are divided into three categories, one of which is healthy. Early and blight image sets each have 1000 photographs, whereas a healthy image set comprises 152 images.

2.8.3 Tomato dataset

There are 10 different classes for tomato plant leaf photos, all of which fall under the category of "healthy." The number of target spot polluted photographs is 1404, while the number of mosaic virus impacted images is 373. Photos of yellow leaf curl virus-infected leaves number 3209, whereas images of tomato plant leaves with bacterial spots number 2127. There are 1000 and 1591 photos in the early blight and healthy categories, respectively. There are 1909 and 952 photos in the late blight and leaf mold categories, respectively. There are 1771 and 1676 photos in the Septoria leaf spot and spider mites image types, respectively.

Table 2.3 Categories of Plant Disease

Plant type	Disease name	No. of images in dataset	Sample image	State of the art techniques used
Bell pepper	Bacterial Spot	997		Fisher vector(FV) extraction, . Gaussian distribution's differentiation, SIFT, SVM, Multi-layer perceptron [105], Laplacian filter, GLCM [97]
	Healthy	1478		
Potato	Early Blight	1000		Fisher vector(FV) extraction, . Gaussian distribution's differentiation, SIFT, SVM, Multi-layer perceptron [105], Laplacian filter, GLCM [97]
	Healthy	152		
	Late Blight	1000		
Tomato	Target Spot	1404		Fisher vector(FV) extraction, . Gaussian distribution's differentiation, SIFT, SVM, Multi-layer perceptron [105], K-Means image segmentation algorithm, GLCM, Support Vector Machine (SVM) and Adaptive Neuro Fuzzy Inference System (ANFIS) [100],], Laplacian filter, GLCM [97], CNN [76]
	Mosaic virus	373		
	Yellow leaf curl virus	3209		
	Bacterial Spot	2127		
	Early Blight	1000		
	Healthy	1591		
	Late Blight	1909		
	Leaf mold	952		
	Septoria leaf spot	1771		
	Two spotted spider mite	1676		

CHAPTER 3

LEAF DISEASE DETECTION BASED ON OPTIMIZED SEGMENTATION AND LAW'S MASK FEATURES

To handle the challenge of leaf disease classification, this chapter suggests the usage of an optimization based segmentation and Law's mask feature extraction. SVM is utilized as a classifier to classify the leaves diseases on the basis of various performance metrics. The analysis is performed on three plant types namely bell pepper (2 classes), potato (3 classes) and tomato (10 classes) on the basis of accuracy, precision and recall.

3.1 PROPOSED METHODOLOGY

3.1.1 Segmentation

Segmentation in images causes the edges to be overlapped. We have considered a number of combinations in which we implemented Fuzzy-C means clustering and K-means clustering. Since, K-means clustering performs better than Fuzzy C-means clustering [12], so for improving the segmentation process, we employed K-means clustering optimized with Grey Wolf Optimization algorithm (GWO) for segmenting the diseased areas in the images. GWO [16] was proposed by Seyedali Mirjalili et al. in 2014. Fig 3.1 shows segmentation process using K-means clustering optimized with GWO. Details of the proposed methodology are given in 3.1.2.

Fig 3.1 Segmentation process using K-means clustering optimized with GWO

3.1.2 Methodology

Fig 3.2 shows the proposed methodology [74] used in this chapter. The work implemented in this chapter achieves our first objective as mentioned in Chapter 2. K-means segmentation optimized with GWO is applied on the input image. Law's mask feature extraction, LBP and GLCM are applied for obtaining features. SVM is utilized for classification and the final prediction 'P' is determined.

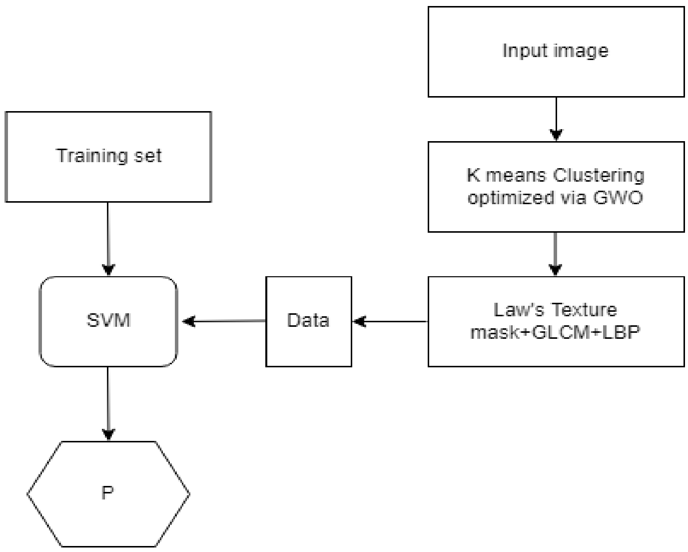


Fig 3.2 Proposed methodology

Following are the steps in detail for the above methodology.

Step 1: The Gaussian distribution is employed on the input photos. This Gaussian distribution normalizes the amount of each image pixel and improves photos by reducing noise.

Step 2: The next step is feature extraction after finding the specific location on the image where the diseases portions exist. Both areas are distinguished by segmentation using clustering, which is enhanced by GWO. In order to overcome the issue of overlapped segmentation, GWO has been employed in the current scenario for enhancing the segmentation procedure. Every wolves' fitness is measured using the objective function, which is a standard established by the analytical technique.

Step 3: Law's feature extraction [42] [17] [29] is applied. Basic convolution kernels are used in this method, and the kernels are applied to identify 2D images. Laws' one-dimensional convolution kernels of a length of three are as follows:

$$L3 = (1, 2, 1) \quad (3.1)$$

$$E3 = (-1, 0, 1) \quad (3.2)$$

$$S3 = (-1, 2, -1) \quad (3.3)$$

These labels stand for Level, Edge, and Spot. The lengths of the kernels can be extended by convolving the pairs of these kernels. For example, if these three kernels are convolved, five new kernels of a length of five are obtained:

$$L5 = (1, 4, 6, 4, 1) \quad (3.4)$$

$$E5 = (-1, -2, 0, 2, 1) \quad (3.5)$$

$$S5 = (-1, 0, 2, 0, -1) \quad (3.6)$$

$$R5 = (1, -4, 6, -4, 1) \quad (3.7)$$

$$W5 = (-1, 2, 0, -2, 1) \quad (3.8)$$

By extending the lengths of the kernels, new kernels referred to as Ripple and Wave are obtained. These sets of one-dimensional kernels are combined to generate two-dimensional kernels, which are then used to analyse 2D images. There are 9 two-dimensional kernels (3x3) for kernels of a length of three:

E3E3 E3L3 E3S3

L3E3 L3L3 L3S3

S3E3 S3L3 S3S3

There are 25 two-dimensional kernels (5x5) for kernels of a length of five:

L5L5 L5E5 L5S5 L5R5 L5W5

E5L5 E5E5 E5S5 E5R5 E5W5

S5L5 S5E5 S5S5 S5R5 S5W5

R5L5 R5E5 R5S5 R5R5 R5W5

W5L5 W5E5 W5S5 W5R5 W5W5

Following are the steps involved in retrieving the above 2-D masks.

a. Convolution is performed between the image $I_{(i,j)}$ and the 2D mask to retrieve the texture image(TI)

For 5x5 mask, $TI_{E_5E_5} = I_{(i,j)} * E_5E_5$…………………...(3.9)

For 3x3 mask, $TI_{E_3E_3} = I_{(i,j)} * E_3E_3$…………………...(3.10)

b. Perform normalization using min-max normalization by passing the image through Texture Energy Measurement (TEM) filters.

For 5x5 mask, $TEM_{ij} = \sum_{u=-5}^{5} \sum_{v=-5}^{5} Normalize(TIi+u, j+v)$……(3.11)

For 3x3 mask, $TEM_{ij} = \sum_{u=-3}^{3} \sum_{v=-3}^{3} Normalize(TIi+u, j+v)$……(3.12)

c. Texture Energy measures (TEMs) denoted as TR are retrieved by blending TEM descriptors.

For 5x5 mask, $TR_{E_5L_5} = \frac{TEME5L5+TEML5E5}{2}$……………..……(3.13)

For 3x3 mask, $TR_{E_3L_3} = \frac{TEME3L3+TEML3E3}{2}$ ………………(3.14)

d. Different statistical features are extracted using GLCM [18] [14]. A few of the statistical features are mentioned below:

Contrast is a measurement of the difference in intensity or grey level between the reference pixel and its neighbour. In GLCM, a large contrasting indicates a high difference in intensity

$Contrast = \sum_{i,j=0}^{N-1} P_{ij}(i-j)^2$……………..…………(3.15) where, P_{ij} = Element i,j of the normalized symmetrical GLCM, N = Number of gray levels in the image as specified by number of levels in under quantization on the GLCM texture page of the variable properties dialog box.

Dissimilarity is the difference in grey level pairs. It's comparable to contrast with the exception that the weights rise in a linear fashion.

$$Dissimilarity = \sum_i \sum_j |i - j| GLCM_{(i,j)} \text{ ………………….(3.16)}$$

Homogeneity determines how near the GLCM element arrangement is to the GLCM diagonally.

$$Homogeneity = \sum_{i,j=0}^{N-1} \frac{Pij}{1+(i-j)2} \text{………………….(3.17)}$$

The orderliness of the pixel values in the frame is measured by the Angular Second Moment (ASM).

$$ASM = G\ X - 1\ i = 0\ G\ X - 1\ j = 0\ \{P_{(i,j)}\}^2 \text{……………..(3.18)}$$

Energy is derived from ASM

$$Entropy = \sum_{i,j=0}^{N-1} \ln(P_{ij}) P_{ij} \text{………………………(3.19)}$$

Step 4: Following that diseased leaf images having 2, 3 and 10 classes for the respective plant types are used to obtain features. Learning is done in several classes employing SVM with “rbf” kernel, and testing is conducted using accuracy, precision and recall.

Algorithm 3.1: Segmentation (Image)

Input: Input Images

Output: Segmented Images

Begin

N← no. of Images

P←i x j Pixels

While (N>0)

Start

$G(P) = \frac{1}{\sqrt{2\pi\sigma^2}} C^{-\frac{X^2}{2\sigma^2}}$……………………………………..............................(3.20)

σ= difference $(P_{i,J-1} - P_{i-1,J})$

End

Define centroid $\{X_1, X_2, \dots \dots \dots \dots, X_n\}$

While (Centroid >0)

Start

Define the population of grey wolves

GW←Centroid

$G_{\propto}$← G(P)

G_{β}← N

G_{δ}← P

Update weights

$w^{n+1} = \frac{w_0\ G_{\propto} + \sum GW}{G_{\propto} + G_{\beta} + G_{\delta}}$…………………………………………………………(3.21)

End

Algorithm 3.2: Law's mask features

Input: Segmented Images

Output: Law mask feature

Begin

While (pixel >0)

Start

Law's mask (5x5) or (3x3)

Normalize features

End

Finish

Algorithm 3.3: Classification (image)

Input: Images

Output: Classified Diseases

Begin

N$\leftarrow$number of images

While (N>0)

Start

Pre-process (N)

Segmentation (N)

Law's mask (N)

LBP (N)

GLCM (N)

Finish

Feature$\leftarrow\{x_1, x_2, \dots \dots \dots, x_n\}$

Label$\leftarrow\{l_1, l_2, \dots \dots \dots, l_n\}$

$T\leftarrow$(features, label)

Train$\leftarrow$ SVM (T)

Test$\leftarrow$ Train (T)

Analyze Accuracy, Precision and Recall

End

3.2 EXPERIMENT AND RESULT ANALYSIS

We employed a variety of evaluation indicators to assess the classification model's performance:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.22),$$

$$Precision = \frac{TP}{TP+FP} \quad (3.23),$$

$$Recall = \frac{TP}{TP+FN} \quad (3.24),$$

where, TP stands for True Positive, TN stands for True Negative, FP stands for False Positive, and FN is for False Negative.

Table 3.1 Experimental Setup

Database	PlantVillage
Leaves	Bell pepper, Potato, Tomato
Number of classes	2,3,10
Classifier	Support Vector Machine
Features	GLCM, LBP, Texture, Law's mask
2-class	Bell Pepper
3-class	Potato
10-class	Tomato
Clustering	K-means with GWO
Evaluation Metrics	Accuracy, Precision, Recall

The experimental configuration for the proposed methodology is shown in Table 3.1. The research employs PlantVillage dataset with three plant types having two, three, and ten classes of diseases tested with accuracy, precision, and recall. We have tested a number of hybrid approaches which are as follows:

- Fuzzy segmentation + GLCM + SVM → This combination employs Fuzzy C-means clustering as the segmentation technique, for feature extraction GLCM is employed while for classification SVM is utilized.
- Fuzzy segmentation + LBP + GLCM + SVM → This combination employs Fuzzy C-means clustering as the segmentation technique, for feature extraction GLCM, LBP is employed while for classification SVM is utilized.
- Optimization Clustering + LBP + GLCM + SVM → This combination employs K-means clustering optimized with Particle Swarm Optimization as the segmentation technique, for feature extraction GLCM, LBP is employed while for classification SVM is utilized.

- Clustering segmentation + LBP + GLCM + SVM → This combination employs K-means clustering as the segmentation technique, for feature extraction GLCM, LBP is employed while for classification SVM is utilized.
- Clustering segmentation optimized with GWO + Law's texture mask (5x5) + LBP + GLCM + SVM → This combination employs K-means clustering optimized with GWO as the segmentation technique, for feature extraction GLCM, LBP and Law's texture mask (5x5) is employed while for classification SVM is utilized.
- Clustering segmentation optimized with GWO + Law's texture mask (3x3) + LBP + GLCM + SVM → This combination employs K-means clustering optimized with GWO as the segmentation technique. For feature extraction, GLCM, LBP and Law's texture mask (3x3) is employed while for classification SVM is utilized.

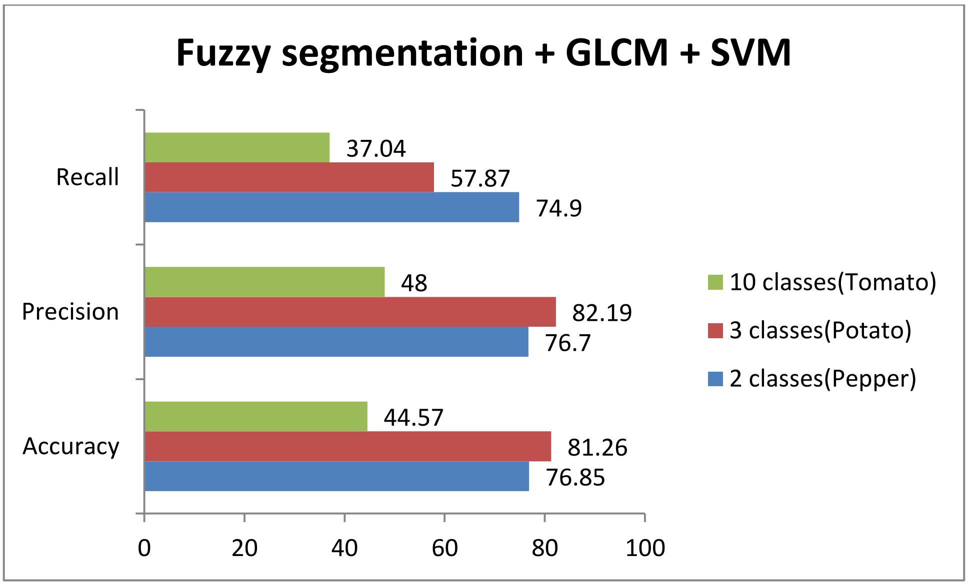


Fig 3.3 Analysis of fuzzy segmentation aggregated with GLCM and SVM

In Fig 3.3, segmentation is performed using Fuzzy C-means clustering. Features are obtained using GLCM algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 76.85%, for potato it is 81.26% and for tomato it is 44.57%. Precision achieved for bell pepper is 76.70%, for potato it is 82.19% and for tomato it is 48%. Recall achieved for bell pepper is 74.90%, for potato it is 57.87% and for tomato it is 37.04%.

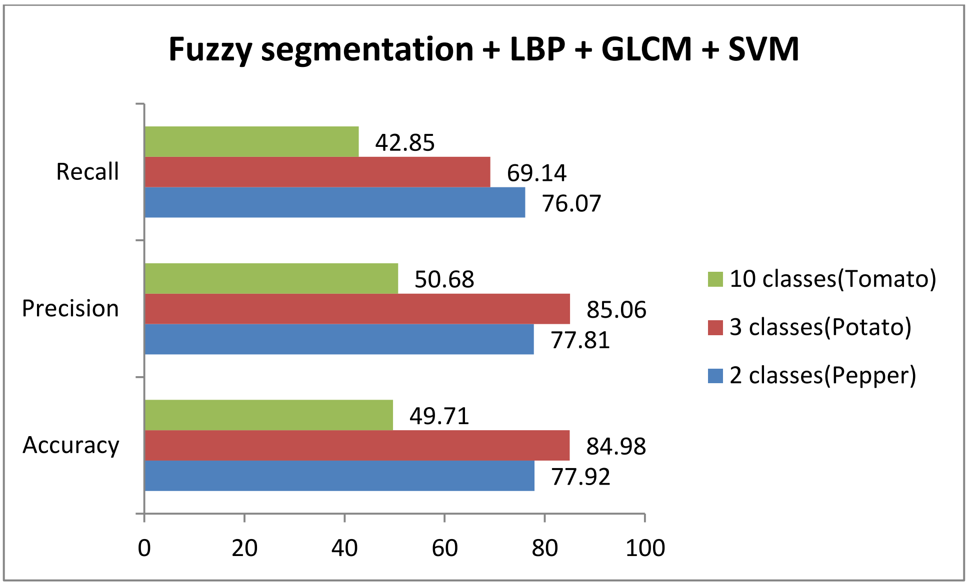


Fig 3.4 Analysis of fuzzy segmentation aggregated with GLCM, LBP and SVM

The fuzzy segmentation with LBP and GLCM is examined in Fig 3.4. Segmentation is performed using Fuzzy C-means clustering. Features are obtained using GLCM and LBP algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 77.92%, for potato it is 84.98% and for tomato it is 49.71%. Precision achieved for bell pepper is 77.81%, for potato it is 85.06% and for tomato it is 50.68%. Recall achieved for bell pepper is 76.07%, for potato it is 69.14% and for tomato it is 42.85%

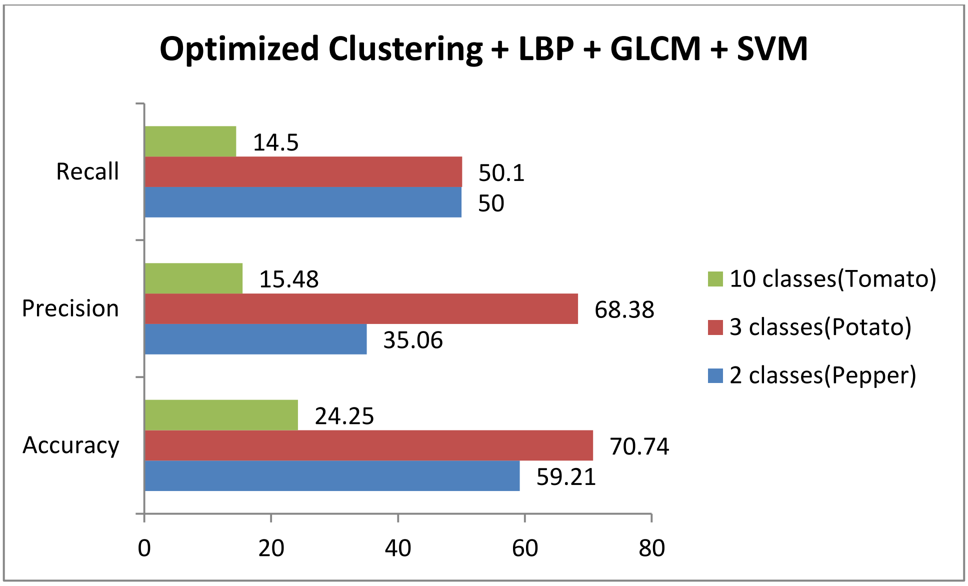


Fig 3.5 Analysis of Optimized Clustering aggregated with GLCM, LBP and SVM

In Fig 3.5, segmentation is performed using Particle Swarm Optimization (PSO). Features are obtained using GLCM and LBP algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 59.21%, for potato it is 70.70% and for tomato it is 24.25%. Precision achieved for bell pepper is 35.06%, for potato it is 68.38% and for tomato it is 15.48%. Recall achieved for bell pepper is 50%, for potato it is 50.10% and for tomato it is 14.50%

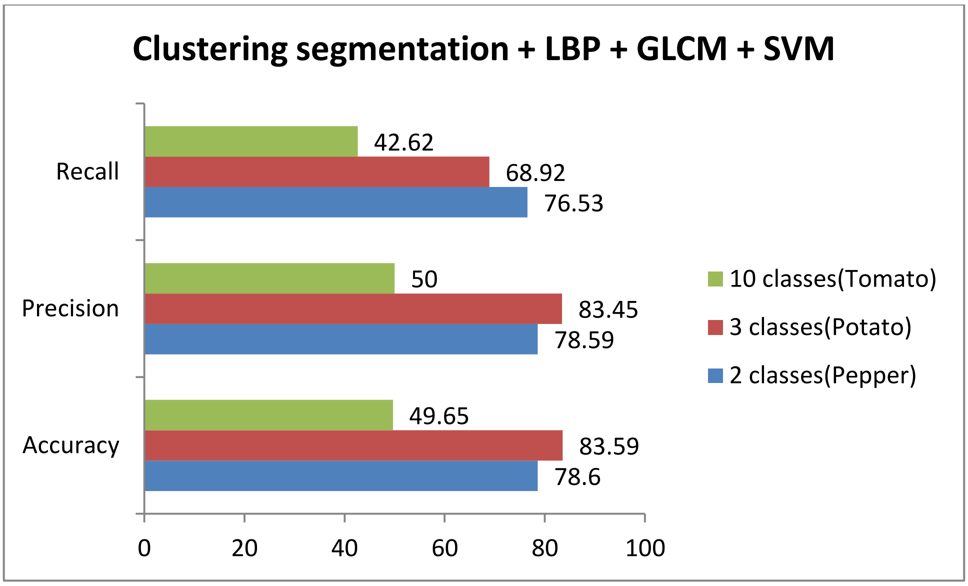


Fig 3.6 Analysis of Clustering Segmentation with GLCM, LBP and SVM

In Fig 3.6, segmentation is performed using K-means clustering. Features are obtained using GLCM and LBP algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 78.60%, for potato it is 83.59% and for tomato it is 49.65%. Precision achieved for bell pepper is 78.59%, for potato it is 83.45% and for tomato it is 50%. Recall achieved for bell pepper is 76.53%, for potato it is 68.92% and for tomato it is 42.62%

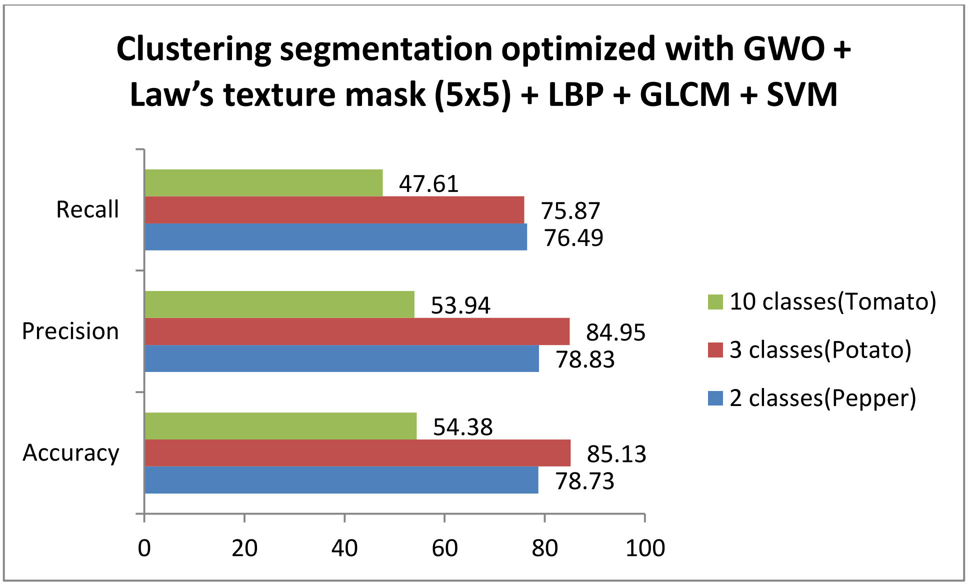


Fig 3.7 Analysis of Clustering Segmentation with Law's mask (5x5) optimized with GWO, GLCM, LBP and SVM

In Fig 3.7, segmentation is performed using K-means clustering which has been optimized with GWO. Features are obtained using 5x5 Law's mask feature extraction, GLCM and LBP algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 78.73%, for potato it is 85.13% and for tomato it is 54.38%. Precision achieved for bell pepper is 78.83%, for potato it is 84.95% and for tomato it is 53.94%. Recall achieved for bell pepper is 76.49%, for potato it is 75.87% and for tomato it is 47.61%

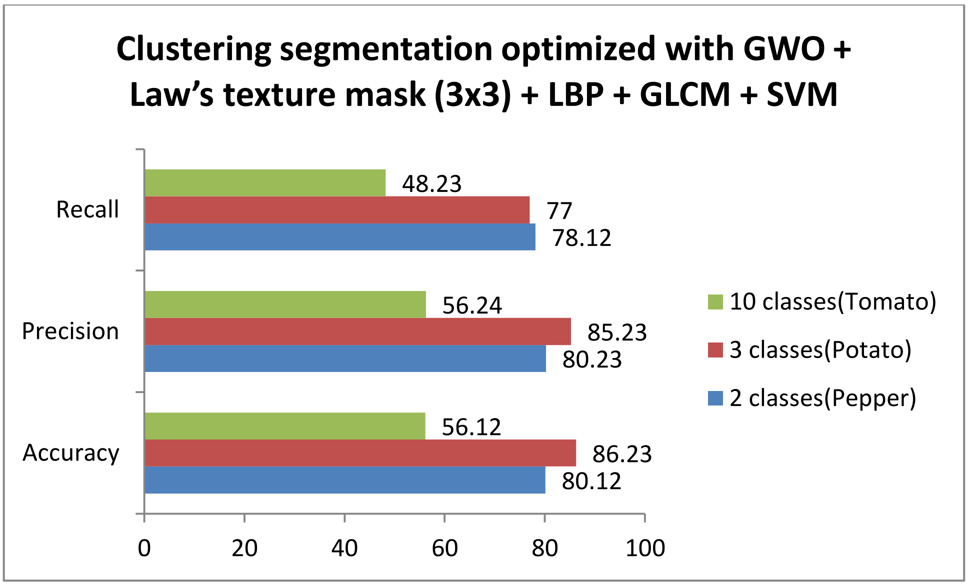


Fig 3.8 Analysis of Clustering Segmentation with Law's mask (3x3) optimized with GWO, GLCM, LBP and SVM

In Fig 3.8, segmentation is performed using K-means clustering optimized with GWO. Features are obtained using 3x3 Law's mask feature extraction, GLCM and LBP algorithm. SVM is used for classification of diseases based on three plant types – bell pepper, potato and tomato having 2, 3, and 10 categories of diseases. It has been observed here that accuracy achieved for bell pepper is 80.12%, for potato it is 86.23% and for tomato it is 56.12%. Precision achieved for bell pepper is 80.23%, for potato it is 85.23% and for tomato it is 56.24%. Recall achieved for bell pepper is 78.12%, for potato it is 77% and for tomato it is 48.23%.

The best combination of techniques found is Clustering Segmentation with Law's mask (3x3) optimized with GWO, GLCM, LBP and SVM. Using this approach, we have calculated the confusion matrix for the individual plant type.

Table 3.2 Confusion matrix for pepper plant type

		Predicted Values	
		Bacterial Spot	**Healthy**
Actual Values	**Bacterial Spot**	350	88
	Healthy	59	246

Table 3.2 shows the confusion matrix for pepper class. The True Positive Rate(TPR) calculated is 0.7812. The False Negative Rate(FNR) is 0.2009. The True Negative Rate(TNR) is 0.8065. The False Positive Rate(FPR) is 0.1934.

Table 3.3 Confusion matrix for potato plant type

		Predicted Values		
		Early Blight	**Healthy**	**Late Blight**
Actual Values	**Early Blight**	300	2	74
	Healthy	3	19	13
	Late Blight	31	25	179

Table 3.3 shows the confusion matrix for potato class. The True Positive Rate(TPR) calculated is 0.77. The False Negative Rate(FNR) is 0.1978. The True Negative Rate(TNR) is 0.8740. The False Positive Rate(FPR) is 0.1259.

Table 3.4 Confusion matrix for tomato plant type

		Predicted Values									
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**0**
	0	440	6	8	9	58	116	136	137	200	328
	1	95	46	56	27	1	28	26	27	0	43
	2	135	12	157	17	25	6	15	17	0	39

Actual Values	**3**	99	32	40	35	16	29	61	21	0	92
	4	114	1	39	15	83	28	39	8	0	45
	5	65	2	17	12	10	52	20	53	15	1
	6	92	4	25	19	4	28	59	41	1	76
	7	56	7	34	27	3	105	59	74	7	6
	8	50	1	4	0	1	56	0	27	22	0
	9	164	12	26	35	4	13	94	9	0	169

Table 3.4 shows the confusion matrix for tomato class. For confusion matrix the classes are represented as numbers as:

0 Bacterial Spot

1 Early Blight

2 Healthy

3 Late Blight

4 Leaf mod

5 Septoria leaf spot

6 Spider mites

7 Target spot

8 Mosaic virus

9 Yellow leaf curl virus

The True Positive Rate(TPR) calculated is 0.48. The False Negative Rate(FNR) is 0.6940. The True Negative Rate(TNR) is 0.7306 The False Positive Rate(FPR) is 0.3555.

3.2.1 Result analysis

Numerous observations are found which are as follows:

- Feature extraction is improved via segmentation because overlapped segmentation, such like fuzzy segmentation, reduces feature distortion.
- The new framework makes use of optimized segmentation to eliminate overlapped segments which improvise the results.

- Accuracy, precision and recall have improved as a result of the feature extraction.
- Law's mask characteristics convoluted kernels in the suggested approach have improved the results.
- The suggested method enhances the feature extraction process, but the process of learning does not boost 10-class categorization because it is dependent on the classifier's learning form.
- As a consequence, the suggested method greatly improves 2-class, 3-class, and 10-class diseases.

3.3 SUMMARY

In this chapter we have optimized the segmentation process and we have selected the best combinations of techniques using Law’s mask features. Law’s mask features have improved the results greatly in the suggested approach. We have observed that the combination using K-means with GWO and a variety of feature extraction techniques such as GLCM and LBP along with the use of Law’s mask have given us the best results in terms of accuracy, precision and recall. We have calculated the average percentage in improvement for the proposed approach (Clustering segmentation optimized with GWO + Law’s texture mask (3x3) + LBP + GLCM + SVM) from other combinations in this. For pepper, the average improvement is 4.89% in accuracy, 9.03% in precision and 6.11% in recall. For potato, the average improvement is 4.25% in accuracy, 3.69% in precision and 10.52% in recall. For tomato, the average improvement is 9.68% in accuracy, 10.52% in precision and 9.43% in recall.

CHAPTER 4
ANALYSIS USING FEATURE REDUCTION

This chapter discusses the implementation of dimensionality reduction in our work in order to improve results with reduced number of features and less time. The two dimensionality reduction techniques utilized for this purpose are Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). The analysis is performed on three plant types namely bell pepper (2 classes), potato (3 classes) and tomato (10 classes) on the basis of accuracy, precision and recall.

4.1 PROPOSED METHODOLOGY

The methodology [90] in this chapter is an extension of the proposed methodology in Chapter 3. In order to reduce the complexity of the model and to attain an improved performance with less number of features, an attempt has been made to reduce the features and then calculate the performance. Fig 4.1 depicts proposed approach flow chart. A brief introduction to the feature reduction techniques is given in 4.1.1. After the application of feature extraction, two feature reduction techniques Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) have been tested on the methodology as in Chapter 3.

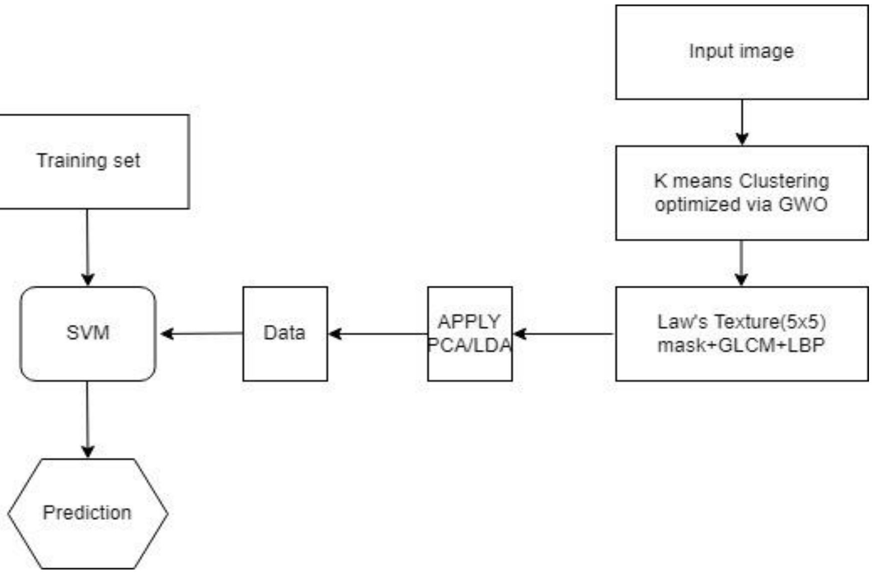


Fig 4.1 Proposed methodology using feature reduction

4.1.1 Dimensionality Reduction

We have considered LDA and PCA for reducing the features and tested the performance. The purpose of LDA is to cast features from a larger space onto a lower-dimensional space, avoiding the dimensionality curse while simultaneously saving resources and reducing dimensionality costs. PCA neglects class labels in favour of identifying the principle components that maximize variance in a given set of data. As a result, it is an unsupervised algorithm. LDA, on the other hand, is a supervised algorithm that aims to discover linear discriminants, or axes that optimize separation between various classes. In multi-class classification tasks, LDA outperforms PCA. PCA, on the other hand, works better when the sample size is limited. Comparisons of classification accuracies used in image classification are good examples.

4.1.2 Principal Component Analysis

Steps in PCA[98][99]:

- Input dataset.
- Calculate the mean for each of the dataset's dimensions.

- Calculate the entire dataset's covariance matrix.

$$Cov(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(Xi - \mu_x)(Yi - \mu_y)\ldots\ldots\ldots\ldots\ldots..(4.1)$$

  Where X and Y are random variables with means μ_x and μ_y.

- Calculate the eigen vectors and eigen values for each eigen vector. If A is a square matrix, v is a vector, and a scalar satisfies Aν = λν, then the eigen value associated with the eigen vector of A is termed λ. The roots of the following characteristic equation are A's eigenvalues.

$$\det(A - \lambda I) = 0\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..(4.2)$$

- To build a d x k dimensional matrix W, sort the eigenvectors by decreasing eigen values and choose k eigen vectors with the largest eigen values.
- To transfer the samples onto the new subspace, use this d x k eigenvector matrix.

4.1.3 Linear Discriminant Analysis

Steps in LDA[67]:

- Calculate the d-dimensional mean vectors for each of the dataset's classes.
- The scatter matrices must be computed (in-between-class and within-class scatter matrix).
- Calculate the scatter matrices' eigen vectors (e1,e2,...,ed) and corresponding eigen values (λ1, λ2,..., λd).
- To build a d x k dimensional matrix W, sort the eigen vectors by decreasing eigen values and choose k eigen vectors with the largest eigen values (where every column represents an eigen vector).
- To transfer the samples onto the new subspace, use this d x k eigen vector matrix. Y=X x W (where X is a n d-dimensional matrix encoding the n samples, and y are the transformed n x k-dimensional samples in the new subspace) is a matrix multiplication that summarizes this.

4.2 EXPERIMENT AND RESULT ANALYSIS

Table 4.1 shows the experimental configuration in this chapter.

Table 4.1 Experimental Configuration

Database	PlantVillage
Leaves	Bell pepper, Potato, Tomato
Number of classes	2,3,10
Classifier	Support Vector Machine
Dimensionality Reduction	PCA, LDA
Features	GLCM, LBP, Texture, Law's mask
2-class	Bell Pepper
3-class	Potato
10-class	Tomato
Clustering	K-means with GWO
Evaluation Metrics	Accuracy, Precision, Recall

4.2.1 Analysis of results before and after applying LDA

We have applied LDA with proposed approach as in Chapter 3. Analysis before reduction and after reduction has been performed. The number of dimensions is reduced from original to C - 1 features, where C is the number of classes. The solver is set to default, that is Singular value decomposition (svd). Number of components is set to class – 1.

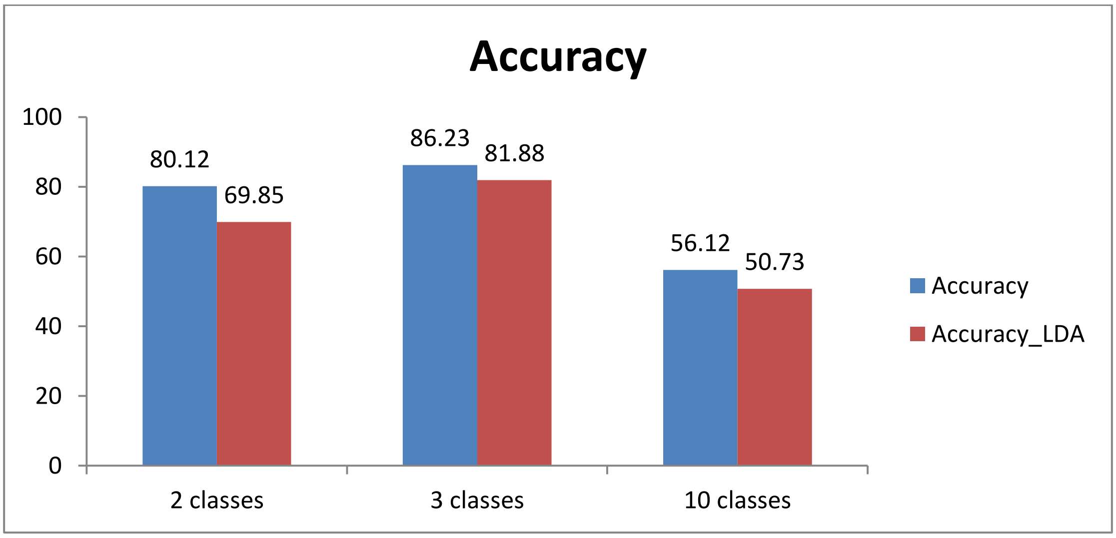


Fig 4.2 Comparison of accuracy without LDA and with LDA

A comparison of accuracy with and without LDA is provided in Fig 4.2. As per Chapter 3, the accuracy of proposed approach was 80.12% for 2 classes, 86.23% for 3 classes, 56.12% for 10 classes. After applying LDA, it has been observed that for 2 classes the accuracy is 69.85%, for 3 classes it is 81.88% while for 10 classes the accuracy obtained is 50.73%.

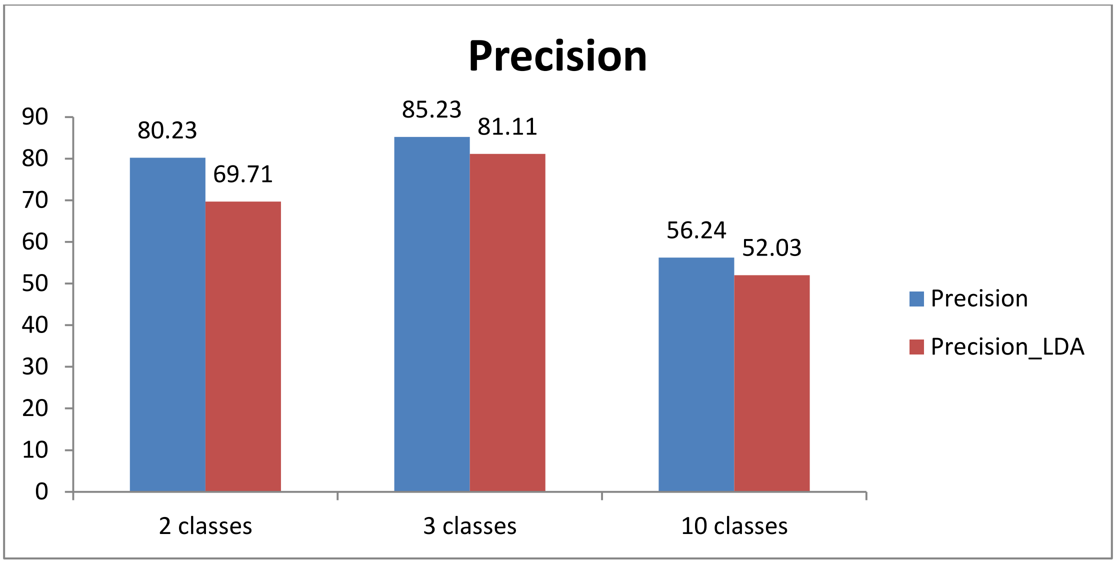


Fig 4.3 Comparison of precision without LDA and with LDA

A comparison of precision with and without LDA is provided in Fig 4.3. As per Chapter 3, the precision of proposed approach was 80.23% for 2 classes, 85.23% for 3 classes, 56.24% for 10 classes. After applying LDA, it has been observed that for 2 classes the precision is 69.71%, for 3 classes it is 81.11% while for 10 classes the precision obtained is 52.03%.

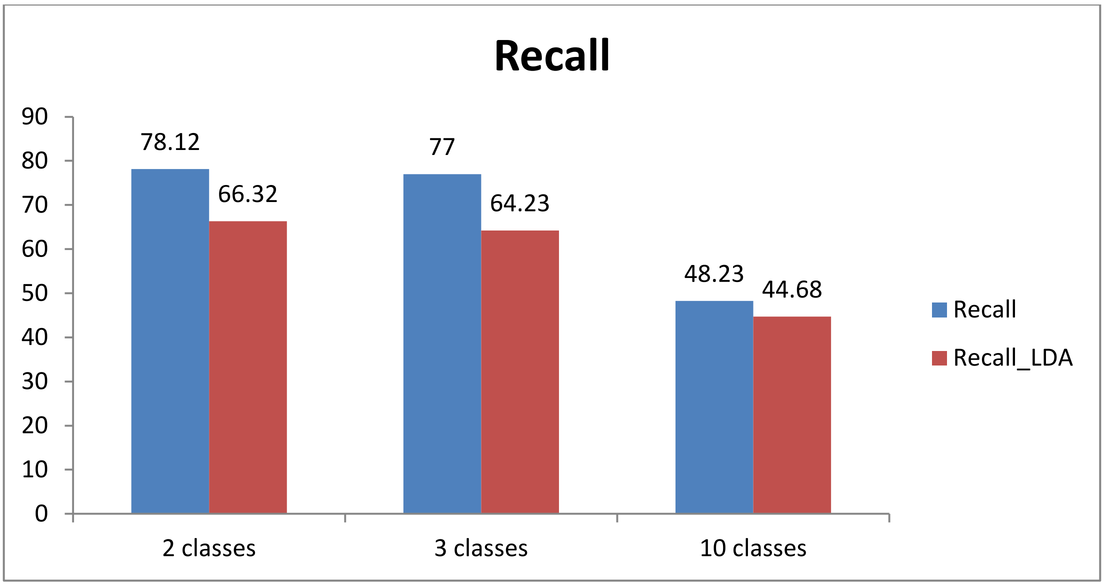


Fig 4.4 Comparison of recall without LDA and with LDA

A comparison of recall with and without LDA is provided in Fig 4.4. As per Chapter 3, the recall of proposed approach was 78.12% for 2 classes, 77.00% for 3 classes, 48.23% for 10 classes. After applying LDA, it has been observed that for 2 classes the recall is 66.32%, for 3 classes it is 64.23% while for 10 classes the recall obtained is 44.68%.

4.2.2 Analysis of results before and after applying PCA

We have implemented 10, 20, 30 and 40 components for PCA. In Fig 4.5, Fig 4.6 and Fig 4.7, PCA-10 refers to PCA employed with 10 components. PCA-20 refers to PCA employed with 20 components. PCA-30 refers to PCA employed with 30 components. PCA-40 refers to PCA employed with 40 components.

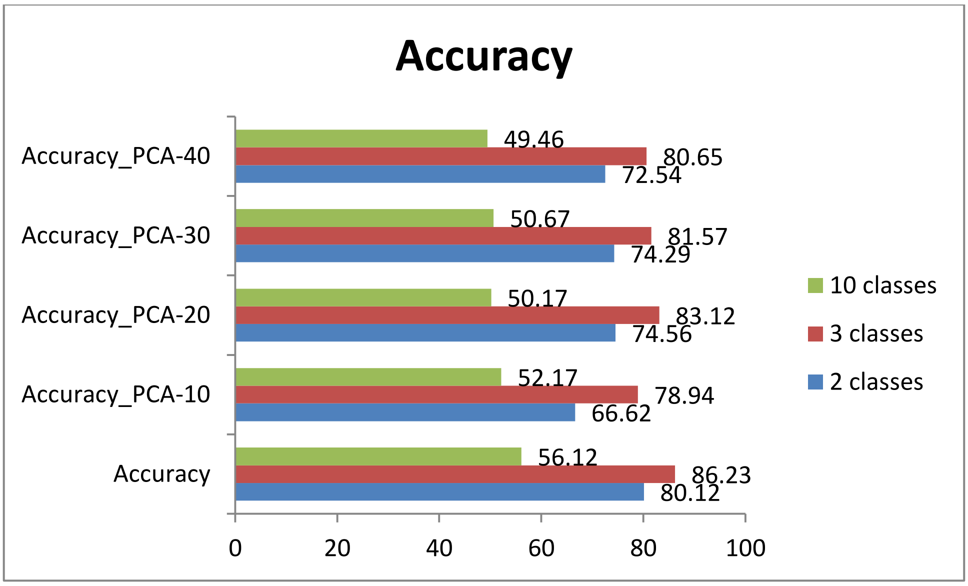


Fig 4.5 Comparison of accuracy without PCA and with PCA

A comparison of accuracy with and without PCA is provided in Fig 4.5. ‘Accuracy’ in Fig 4.5 indicates the accuracy before using PCA (as in Chapter 3). Accuracy_PCA-10 indicates accuracy of proposed approach after applying PCA with 10 components. Accuracy_PCA-20 indicates accuracy of proposed approach after applying PCA with 20 components. Accuracy_PCA-30 indicates accuracy of proposed approach after applying PCA with 30 components. Accuracy_PCA-40 indicates accuracy of proposed approach after applying PCA with 40 components. We employed an SVM classifier with 2, 3, and 10 classes of diseased leaf images to get our results. Among the reduced features, it has been observed that for bell pepper (2 classes), the maximum accuracy is achieved by PCA-20 (74.56%). For potato (3 classes), the maximum accuracy is achieved by PCA-20 (83.12%). For tomato (10 classes), the maximum accuracy is achieved by PCA-10 (52.17%).

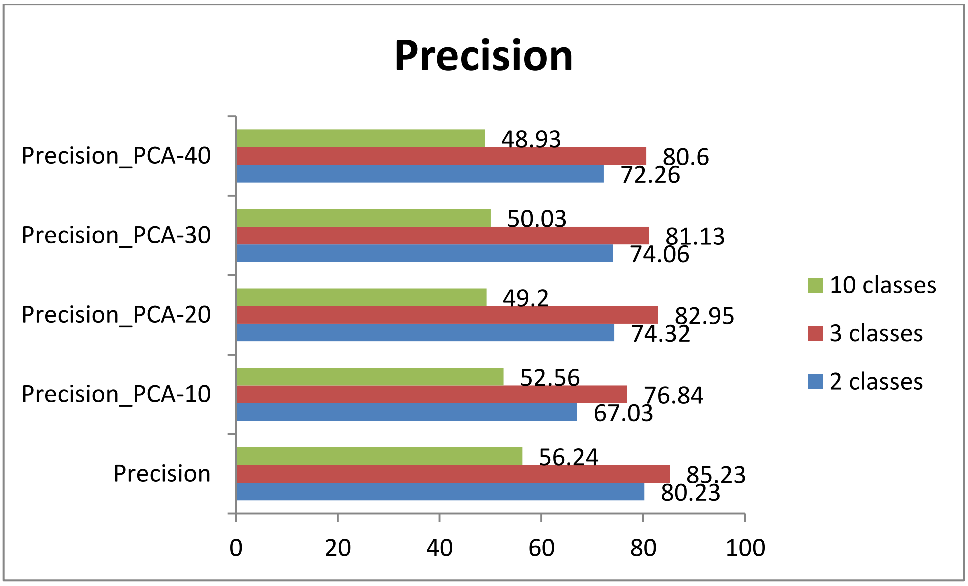


Fig 4.6 Comparison of precision without PCA and with PCA

A comparison of precision with and without PCA is provided in Fig 4.6. 'Precision' in Fig 4.6 indicates the precision before using PCA (as in Chapter 3). Precision_PCA-10 indicates precision of proposed approach after applying PCA with 10 components. Precision_PCA-20 indicates precision of proposed approach after applying PCA with 20 components. Precision_PCA-30 indicates precision of proposed approach after applying PCA with 30 components. Precision_PCA-40 indicates precision of proposed approach after applying PCA with 40 components. We employed an SVM classifier with 2, 3, and 10 classes of sick leaf pictures to get our results. Among the reduced features, it has been observed that for bell pepper (2 classes), the maximum precision is achieved by PCA-20 (74.32%). For potato (3 classes), the maximum precision is achieved by PCA-20 (82.95%). For tomato (10 classes), the maximum precision is achieved by PCA-10 (52.56%).

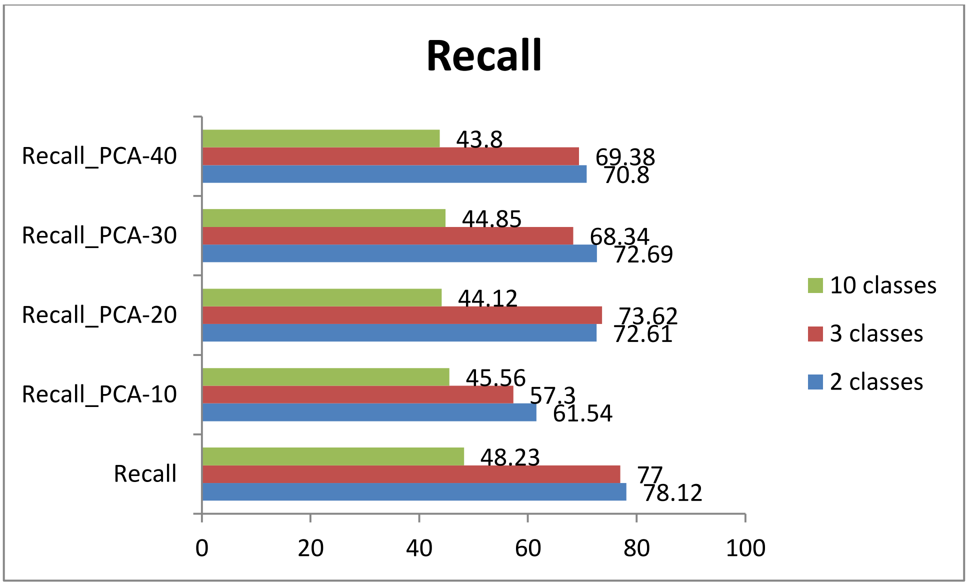


Fig 4.7 Comparison of recall without PCA and with PCA

A comparison of recall with and without PCA is provided in Fig 4.7. 'Recall' in Fig 4.7 indicates the recall before using PCA (as in Chapter 3). Recall_PCA-10 indicates recall of proposed approach after applying PCA with 10 components. Recall_PCA-20 indicates recall of proposed approach after applying PCA with 20 components. Recall_PCA-30 indicates recall of proposed approach after applying PCA with 30 components. Recall_PCA-40 indicates recall of proposed approach after applying PCA with 40 components. We employed an SVM classifier with 2, 3, and 10 classes of sick leaf pictures to get our results. Among the reduced features, it has been observed that for bell pepper (2 classes), the maximum recall is achieved by PCA-30 (72.69%). For potato (3 classes), the maximum recall is achieved by PCA-20 (73.62%). For tomato (10 classes), the maximum recall is achieved by PCA-10 (45.56%).

From the above results, it can be concluded that for bell pepper and potato, PCA with 20 components has given the close results to the proposed approach, while for tomato PCA with 10 components has given the close results to proposed approach.

4.3 SUMMARY

We have tested LDA and PCA with the proposed approach as in Chapter 3. We have observed that PCA has shown better results using all performance metrics as compared to LDA. However, it has also been observed that the proposed approach as discussed in Chapter 3 (without feature reduction) has shown the highest accuracy, precision and recall as compared to results obtained with reduced number of features. A significant reduction in accuracy, precision and recall was observed. LDA has shown 10.27% reduction in accuracy for pepper, 4.35% reduction in accuracy for potato and 5.39% reduction in accuracy for tomato. It has shown 10.52% reduction in precision for pepper, 4.12% reduction in precision for potato and 4.21% reduction in precision for tomato. Also, it has shown 11.8% reduction in recall for pepper, 12.77% reduction in recall for potato and 3.55% reduction in recall for tomato. PCA has shown 8.12% reduction in accuracy for pepper, 5.16% reduction in accuracy for potato and 5.51% reduction in accuracy for tomato. It has shown 8.32% reduction in precision for pepper, 4.85% reduction in precision for potato and 6.06% reduction in precision for tomato. Also, it has shown 8.71% reduction in recall for pepper, 9.84% reduction in recall for potato and 3.65% reduction in recall for tomato. So, we can conclude that dimensionality reduction reduces the performance in our work.

CHAPTER 5

DESIGNING ENSEMBLE CLASSIFIER

This chapter focuses on using ensemble classifier in conjunction with hybrid of Law's mask, LBP, GLCM to boost classification results. The suggested method demonstrates that an ensemble of the selected classifiers can outperform individual classifiers. Because ensemble classification has shown to be more accurate, the features used are also important in achieving the best results. The studies were carried out on the PlantVillage dataset's diseased leaf images of bell pepper, potato, and tomato.

In this chapter, we have employed the concept of ensemble learning [92]. Ensemble learning is typically used to boost a model's classification or prediction accuracy. When contrasted to solo classifiers, ensemble learning [91] [94] [89] approaches have consistently demonstrated higher performance. By integrating different models, ensemble learning improves machine learning results. When compared to a single model, this method offers for improved predictive performance. The basic concept is to train a group of classifiers (experts) and then let them vote (judge). We have utilized the concept of majority voting here. A Voting Classifier is a machine learning model that learns from an ensemble of models and estimates an output (label) based on highest probability of the output being the chosen class. Ensemble learning approaches are frequently described in terms of weak and strong learners. Our objective is to build a powerful learner from the predictions of several weak learners. Classifiers such as Random Forest (RF), Artificial Neural Network (ANN), Support Vector Machine (SVM), K Nearest Neighbor (KNN), Logistic regression (LR), and Naïve Bayes (NB) were used in the suggested method.

5.1 PERFORMANCE TEST OF INDIVIDUAL CLASSIFIERS

In this section, performance of the individual classifiers (ANN, SVM, KNN, LR and NB) has been analysed. These classifiers have been used in conjunction with the proposed approach in Chapter 3.

5.1.1 Artificial Neural Network

These are built on the notion of human brains and are made up of thousands of simple computer elements that work at the same time. ANN [26] can be used to extract and classify visual characteristics in computer vision applications. The neural networks have the ability to respond to their own responses through dynamic weight adjustment in order to perform activities such as information processing. A multilayer perceptron (MLP) is a type of feedforward artificial neural network (ANN). The name MLP can be used to describe any feedforward ANN, or it can relate to networks made up of many layers of perceptrons. In this work, the neural network utilizes the hidden layer size as 100 neurons with 'the rectified linear unit' function (returns f(x) = max(0, x)) as the activation function. The batch size is 200. The learning rate is set to constant and maximum number of iterations is set is to 200. The model is trained using stochastic gradient based optimizer.

5.1.2 Support Vector Machines

SVM [52] is a supervised learning approach that may be applied for both classification and regression problems. Because of its superior recognition rate when contrasted to other classification techniques, it has brought a large amount of researchers from all around the world. It is mostly used to classify two classes in a high-dimensional feature space using a hyperplane. The basic goal is to find a hyperplane in an n-dimensional space which could classify the vertices into related categories. The hyperplane can be written as:

$$w.x + b = 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.1)$$

where, w is the weight vector, which is perpendicular to the hyperplane. The parameter b stands for bias or threshold. For multi-class classification, SVM uses the "one-versus-rest" technique. There are a total of n classes * (n classes - 1) / 2 classifiers built, each of which trains data from two classes. The decision function shape option allows you to convert the outcomes of "one-versus-one" classifiers to a "one-vs-rest" decision function of shape to give a consistent interface with other classifiers. SVM used in this work utilizes ‘rbf’ kernel with regularization parameter set to 1. Decision function shape is set to ‘ovr’.

5.1.3 K- nearest neighbor

For classification problems, KNN [44] is also a supervised learning model. It doesn't do any kind of learning. It is most commonly used to classify a data point based on the classification of its neighbors. In this research, number of neighbors utilized is 3. The weight function used for prediction is set to default value ‘uniform’. It means each neighborhood's points are equally weighted. The algorithm that was used to calculate the nearest neighbours is set to ‘auto’. Based on the values supplied to the fit method, 'auto' will seek to discover the most suited algorithm out of BallTree, KDTree and brute force search. The distance metric used is ‘Euclidean’ distance.

5.1.4 Logistic Regression

A supervised learning approach for categorization is logistic regression [71]. It is mostly employed in the field of data analytics. It is based on the concept of likelihood and is used for predictive modelling. The Sigmoid function is a sophisticated cost function used in logistic regression. This function is used to convert probabilities into forecasts. The logistic function (sometimes known as the sigmoid) is employed, and it is defined as follows:

$$f(x) = \frac{1}{1+e^{-x}} \quad \text{……………………………….(5.2)}$$

where, x is the function's input value. In logistic regression, the weighted sum is used instead of x.

The norm of the penalty is set to l2. Inverse of regularization strength is set to 1.0. Weights associated with class are set to 'none'. The solver algorithm used for optimization problem is set to default 'lbfgs' and the maximum number of iterations is set to default, that is 100. The one-vs-rest (OvR) strategy is used by the training algorithm in the multiclass case. Here, in our work, we used the 'multi-class' case. The value is set to 'ovr' for multi-class classification.

5.1.5 Naïve Bayes

The Bayesian classifiers are probabilistic classifiers based on Bayes' theorem [56]. The "naive" assumption of conditional independence between every pair of features given the value of the class variable is used in Naive Bayes methods, which are a collection of supervised learning algorithms based on Bayes' theorem. It presupposes that the traits are independent of one another.

5.1.6 Random Forest

The Random Forest is a decision tree composition that has shown to be quite useful in a variety of disciplines throughout the years. The total number of trees in the classifier is set to default, that is 100. To estimate the quality of split, criterion is set to default, that is 'gini'. The maximum depth of tree is set to 2. To split an internal node, the minimum number of samples is set to default, that is 2. The bare minimum of samples that must be present at a leaf node is set to default, that is 1.

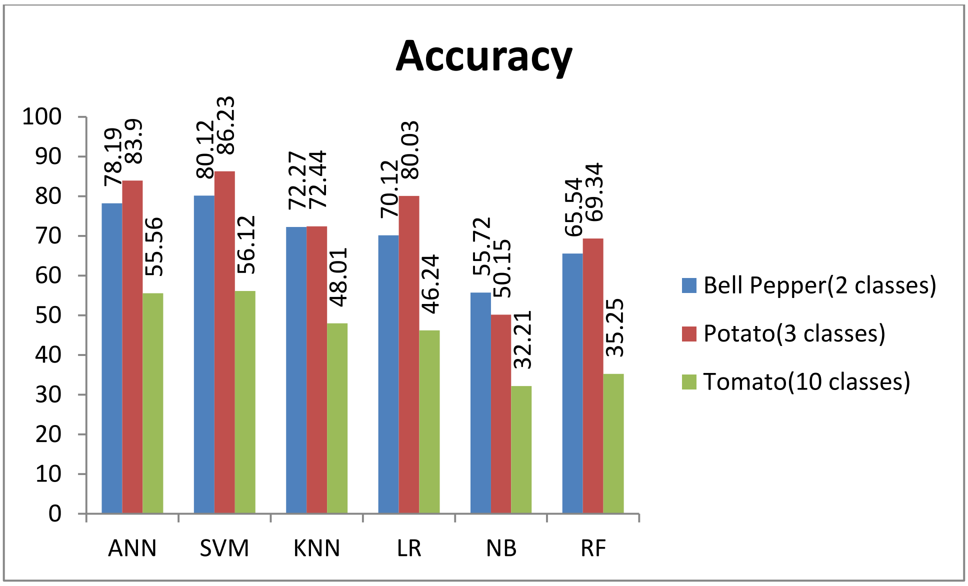


Fig 5.1 Accuracy of individual classifiers

Fig 5.1 shows the accuracy of all classifiers individually. ANN shows 78.19% accuracy for bell pepper, 83.9% for potato and 55.56% for tomato. SVM shows 80.12% accuracy for bell pepper, 86.23% for potato and 56.12% for tomato. KNN shows 72.27% accuracy for bell pepper, 72.44% for potato and 48.01% for tomato. LR shows 70.12% accuracy for bell pepper, 80.03% for potato and 46.24% for tomato. NB shows 55.72% accuracy for bell pepper, 50.15% for potato and 32.21% for tomato. RF shows 65.54% accuracy for bell pepper, 69.34% for potato and 35.25% for tomato.

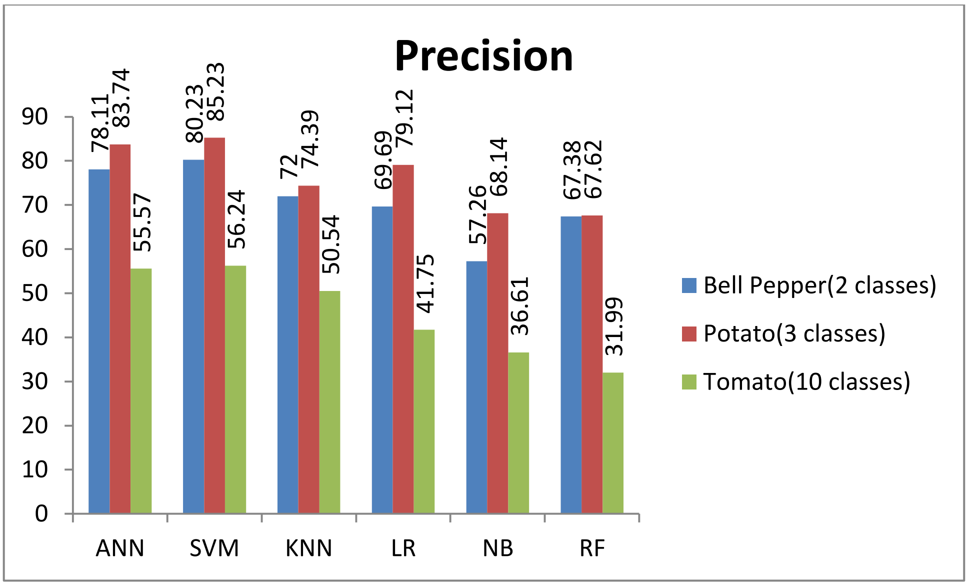


Fig 5.2 Precision of individual classifiers

Fig 5.2 shows the precision of all classifiers individually. ANN shows 78.11% precision for bell pepper, 83.74% for potato and 55.57% for tomato. SVM shows 80.23% precision for bell pepper, 85.23% for potato and 56.24% for tomato. KNN shows 72.00% precision for bell pepper, 74.39% for potato and 50.54% for tomato. LR shows 69.69% precision for bell pepper, 79.12% for potato and 41.75% for tomato. NB shows 57.26% precision for bell pepper, 68.14% for potato and 36.61% for tomato. RF shows 67.38% precision for bell pepper, 67.62% for potato and 31.99% for tomato.

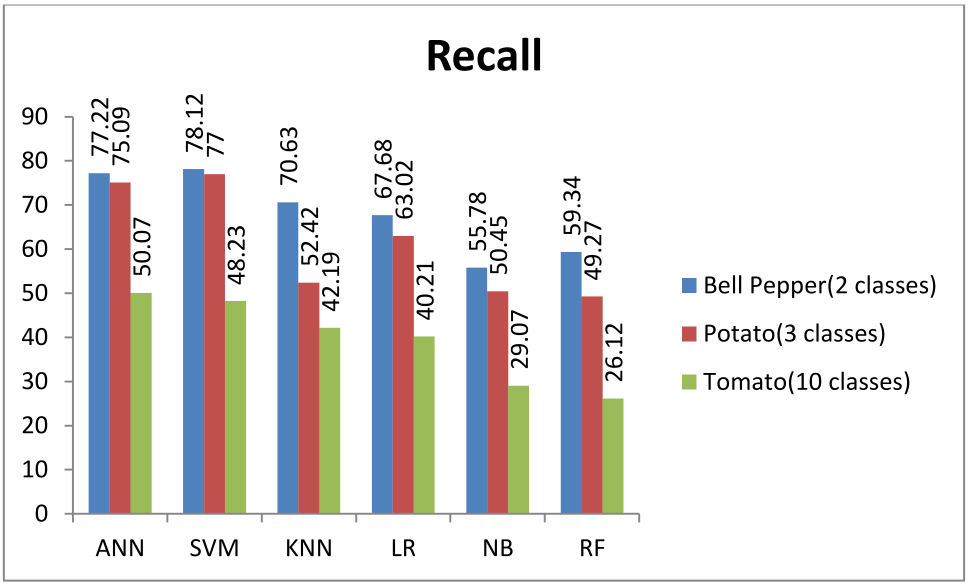


Fig 5.3 Recall of individual classifiers

Fig 5.3 shows the recall of all classifiers individually. ANN shows 77.22% recall for bell pepper, 75.09% for potato and 50.07% for tomato. SVM shows 78.12% recall for bell pepper, 77.00% for potato and 48.23% for tomato. KNN shows 70.63% recall for bell pepper, 52.42% for potato and 42.19% for tomato. LR shows 67.68% recall for bell pepper, 63.02% for potato and 40.21% for tomato. NB shows 55.78% recall for bell pepper, 50.45% for potato and 29.07% for tomato. RF shows 59.34% recall for bell pepper, 49.27% for potato and 26.12% for tomato.

From Fig 5.1, Fig 5.2 and Fig 5.3, we also have another important finding that SVM has shown the best results while Naïve Bayes has shown the least results in terms of accuracy, precision and recall.

5.2 CHOOSING THE BEST ENSEMBLE CLASSIFIER

To test the performance of ensemble classifiers, we have implemented the following combinations with proposed methodology in Chapter 3. We have not included RF in these combinations since RF is itself an ensemble of decision trees which will increase the complexity of the classifier. To choose the ensemble, we have divided the combinations in 3 parts – ensemble comprising 2 classifiers, ensemble comprising 3 classifiers, ensemble comprising 4 classifiers and ensemble comprising all classifiers. Fig 5.4 shows the different combinations for choosing the best ensemble.

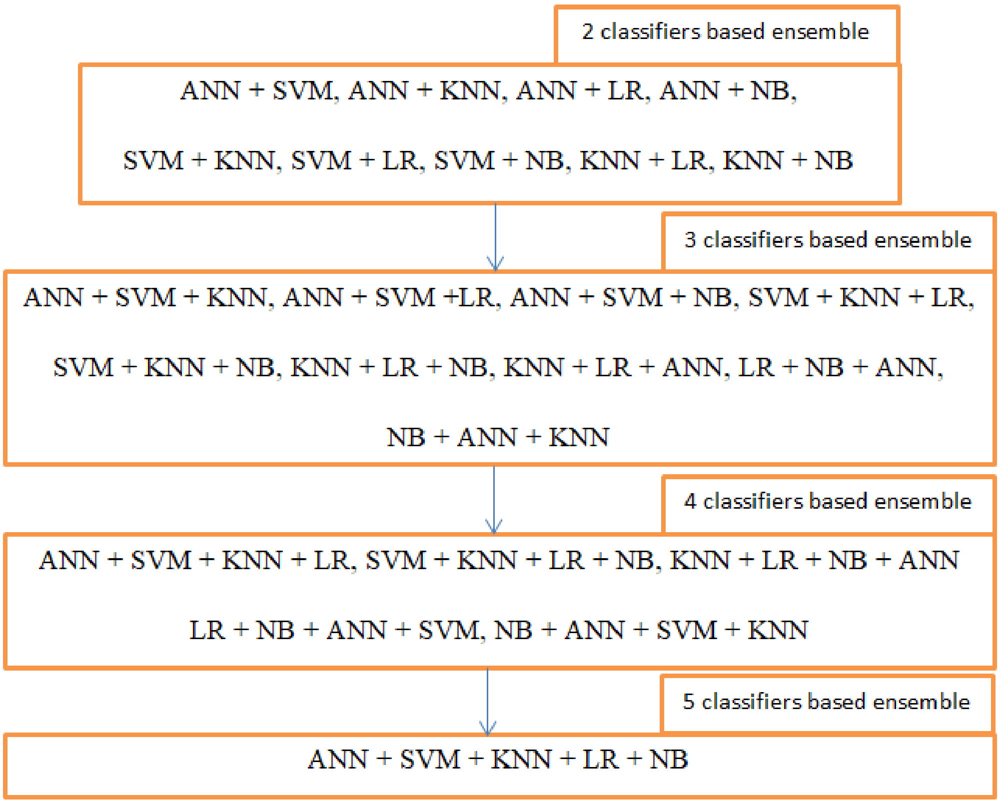


Fig 5.4 Combinations of ensemble classifiers

5.2.1 Two classifiers based ensemble classifier

This comprises ANN + SVM, ANN + KNN, ANN + LR, ANN + NB, SVM + KNN, SVM + LR, SVM + NB, KNN + LR, KNN + NB

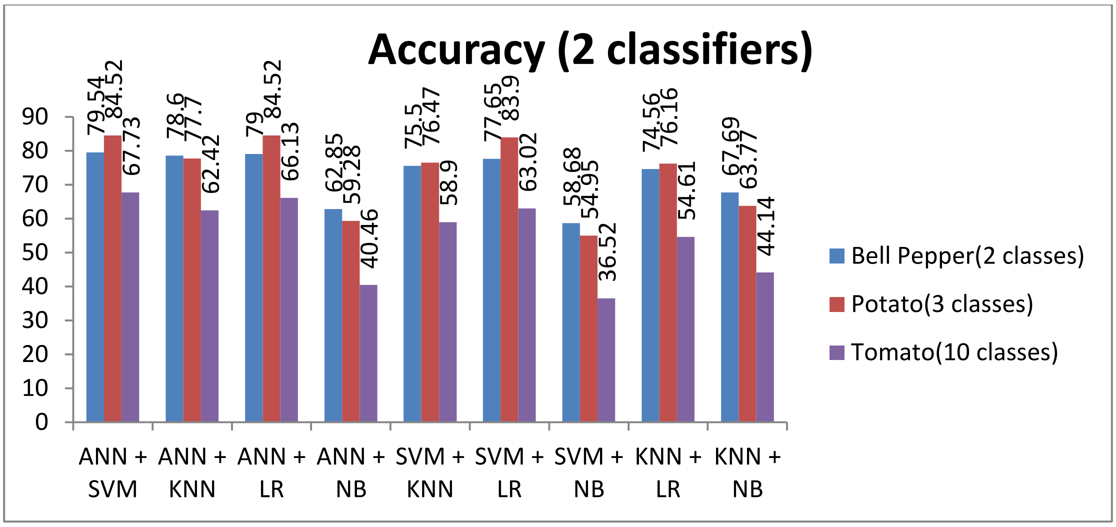


Fig 5.5 Accuracy for two classifiers based ensemble classifier

As per Fig 5.5, the combination of ANN and SVM has shown the best accuracy for all the plant types. 79.54% was found for bell pepper, 84.52% for potato and 67.73% for tomato.

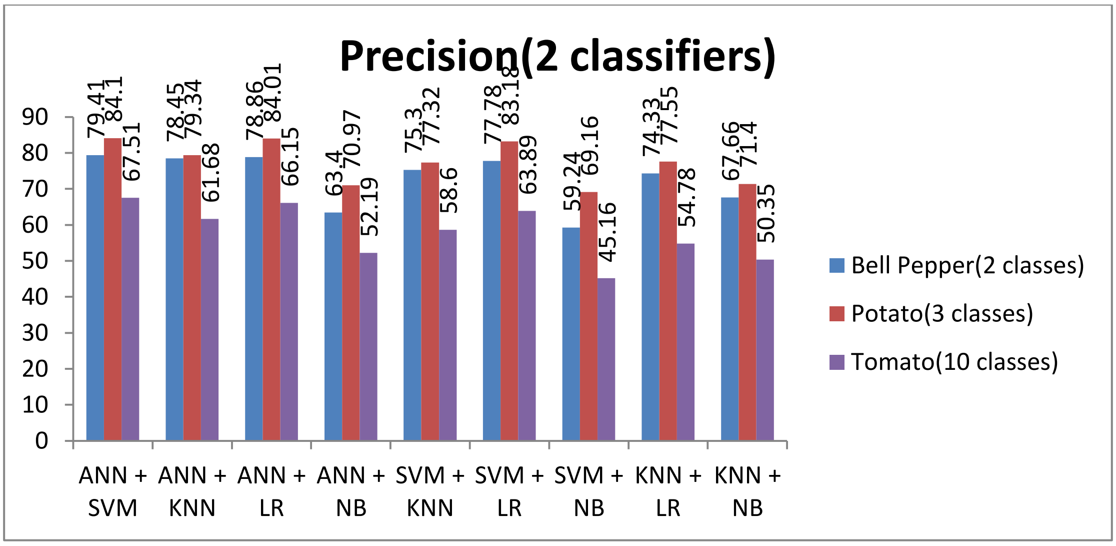


Fig 5.6 Precision for two classifiers based ensemble classifier

As per Fig 5.6, the combination of ANN and SVM has shown the best precision for all the plant types. 79.41% was found for bell pepper, 84.10% for potato and 67.51% for tomato.

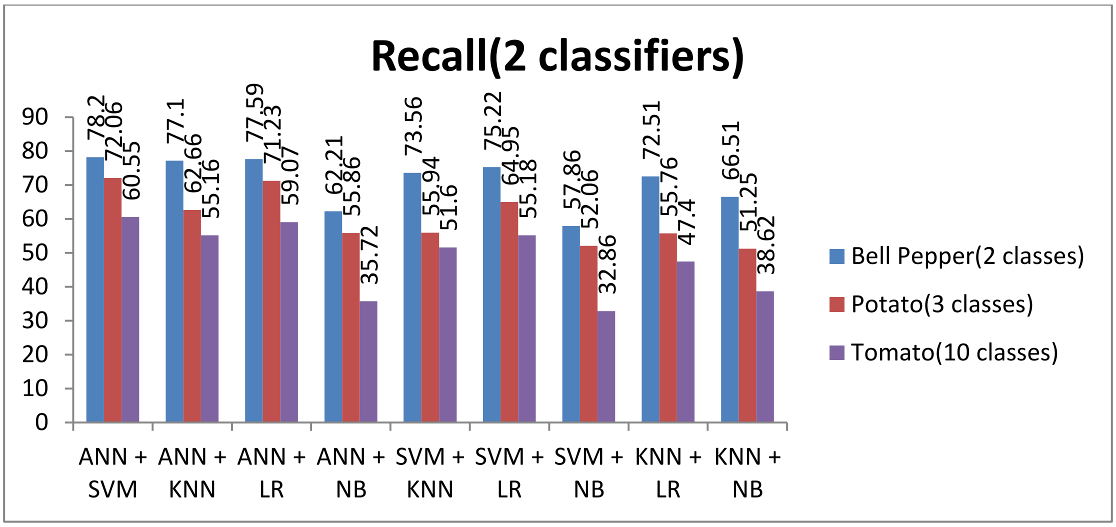


Fig 5.7 Recall for two classifiers based ensemble classifier

As per Fig 5.7, the combination of ANN and SVM has shown the best recall for all the plant types. 78.20% was found for bell pepper, 72.06% for potato and 60.55% for tomato.

5.2.2 Three classifiers based ensemble classifier

This comprises ANN + SVM + KNN, ANN + SVM +LR, ANN + SVM + NB, SVM + KNN + LR, SVM + KNN + NB, KNN + LR + NB, KNN + LR + ANN, LR + NB + ANN, NB + ANN + KNN

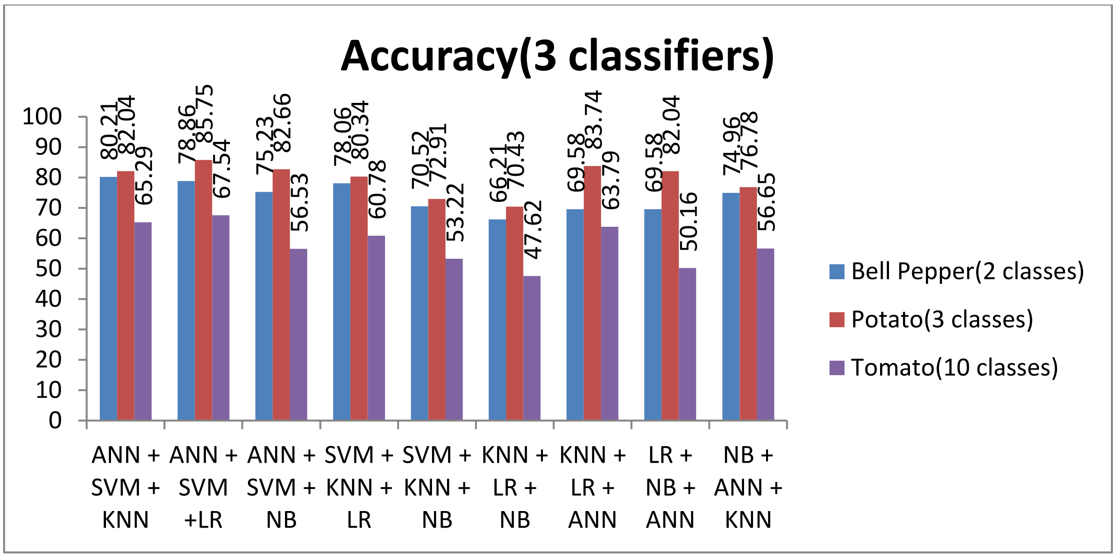


Fig 5.8 Accuracy for three classifiers based ensemble classifier

As per Fig 5.8, the combination of ANN, SVM and KNN has shown the best accuracy for all the plant types. 80.21% was found for bell pepper, 82.04% for potato and 65.29% for tomato.

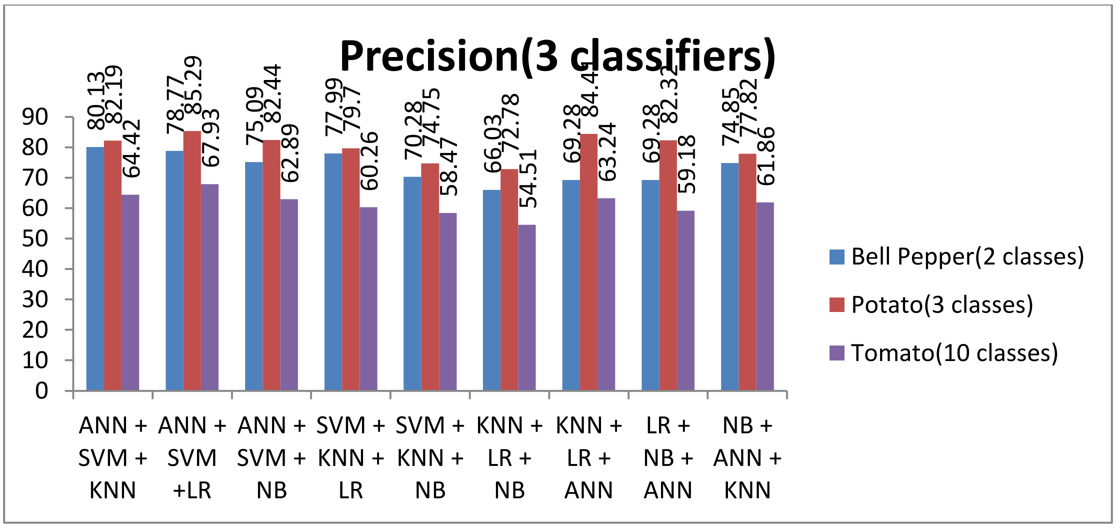


Fig 5.9 Precision for three classifiers based ensemble classifier

As per Fig 5.9, the combination of ANN, SVM and KNN has shown the best precision for all the plant types. 80.13% was found for bell pepper, 82.19% for potato and 64.42% for tomato.

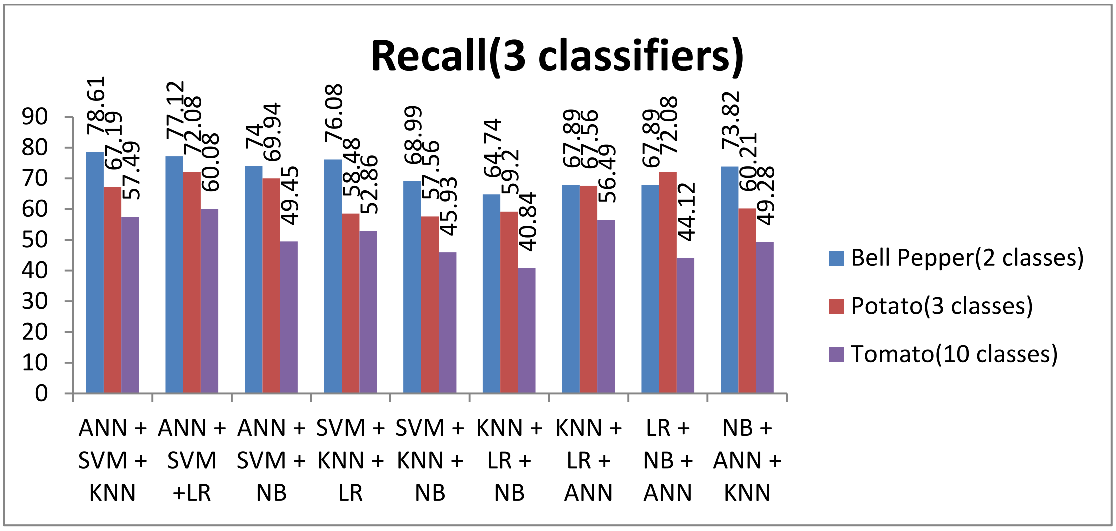


Fig 5.10 Recall for three classifiers based ensemble classifier

As per Fig 5.10, the combination of ANN, SVM and KNN has shown the best recall for all the plant types. 78.61% was found for bell pepper, 67.19% for potato and 57.49% for tomato.

5.2.3 Four and five classifiers based ensemble classifier

For a combination of four classifiers, this comprises ANN + SVM + KNN + LR, SVM + KNN + LR + NB, KNN + LR + NB + ANN, LR + NB + ANN + SVM, NB + ANN + SVM + KNN and for five classifiers, we have implemented an ensemble of all the classifiers.

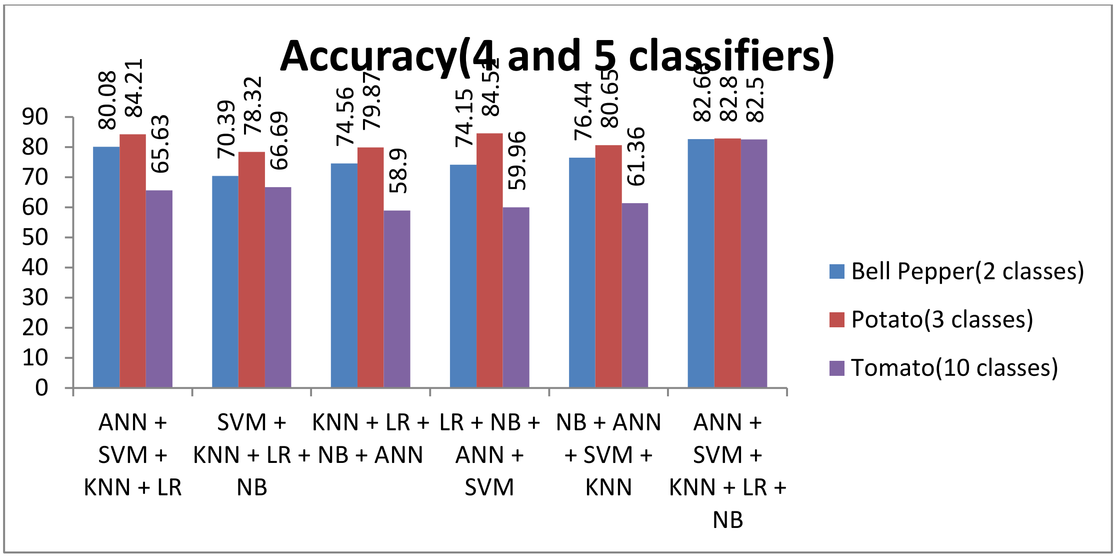


Fig 5.11 Accuracy for four and five classifiers based ensemble classifier

As per Fig 5.11, when we combine four classifiers, we observe that the combination of ANN, SVM, KNN and LR has shown the best accuracy for all the plant types. 80.08% was found for bell pepper, 84.21% for potato and 65.63% for tomato. But when we combine all the classifiers, we get robust results for all the plant types. An accuracy of 82.66% is observed for bell pepper, 82.80% for potato and 82.50% for tomato.

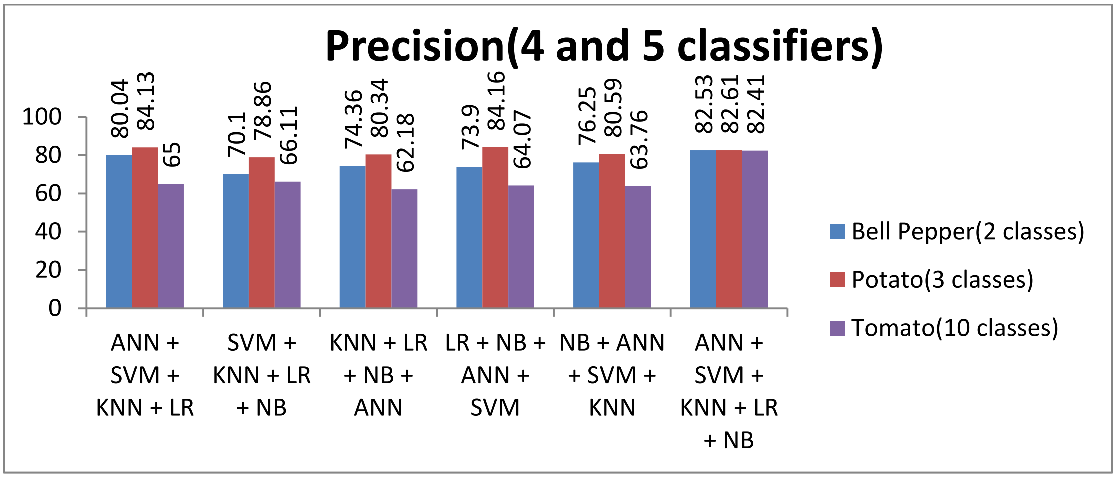


Fig 5.12 Precision for four and five classifiers based ensemble classifier

As per Fig 5.12, when we combine four classifiers, we observe that the combination of ANN, SVM, KNN and LR has shown the best precision for all the plant types. 80.04% was found for bell pepper, 84.13% for potato and 65.00% for tomato. But when we combine all the classifiers, we get improved precision for all the plant types. A precision of 82.53% is observed for bell pepper, 82.61% for potato and 82.41% for tomato.

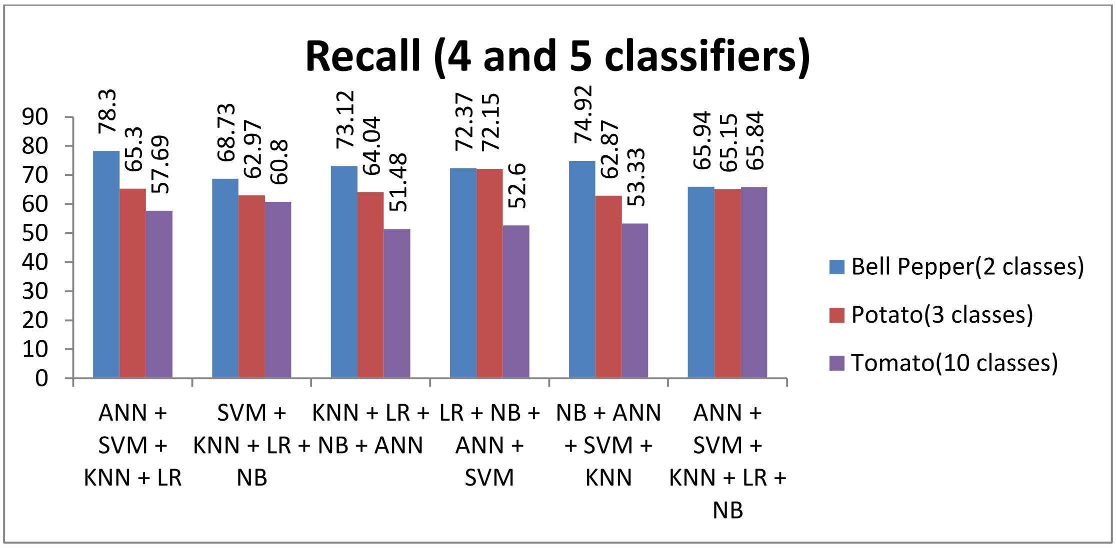


Fig 5.13 Recall for four and five classifiers based ensemble classifier

As per Fig 5.13, when we combine four classifiers, we observe that the combination of ANN, SVM, KNN and LR has shown the best recall for all the plant types. 78.30% was found for bell pepper, 65.30% for potato and 57.69% for tomato. But when we combine all the classifiers, a recall of 65.94% for bell pepper, 65.15% for potato and 65.84% for tomato was observed.

5.3 PROPOSED METHODOLOGY

From the results analyzed in Section 5.2, it was clear that the ensemble of all the classifiers has shown robust results for all plant types in terms of accuracy, precision and recall. So, we have proposed the methodology based on the same. Fig 5.14 refers to the proposed methodology. The input image undergoes K-means [53] clustering segmentation that has been improvised utilizing GWO [16]. Following this the feature extraction is employed using Law's Texture features [42] [17] [29], GLCM [18] [14], LBP [35]. The generated data is fed into the classifier model and predictions (P_1, P_2, P_3,…., P_f) are made regarding the possible disease. Using the concept of voting, the final prediction P_f is the actual output (disease).

Following are the steps in detail for the above methodology.

Step 1: The Gaussian distribution is employed on the input images. This Gaussian distribution normalizes the amount of each image pixel and improves images by reducing noise.

Step 2: The next step is feature extraction after finding the specific location on the image where the diseased portions exist. Both areas are distinguished by segmentation using clustering, which is enhanced by GWO. In order to overcome the issue of overlapped segmentation, GWO has been employed in the current scenario for enhancing the segmentation procedure. Every wolves' fitness is measured using the objective functions, which is a standard established by the analytical technique.

Step 3: Law's feature extraction is applied. Basic convolution kernels are used in this method, and the kernels are applied to identify 2D images. Laws' one-dimensional convolution kernels of a length of three are as follows:

$$L3 = (1, 2, 1) \quad (5.3)$$

$$E3 = (-1, 0, 1) \quad (5.4)$$

$$S3 = (-1, 2, -1) \quad (5.5)$$

These labels stand for Level, Edge, and Spot. The lengths of the kernels can be extended by convolving the pairs of these kernels. For example, if these three kernels are convolved, five new kernels of a length of five are obtained:

$$L5 = (1, 4, 6, 4, 1) \quad (5.6)$$

$$E5 = (-1, -2, 0, 2, 1) \quad (5.7)$$

$$S5 = (-1, 0, 2, 0, -1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (5.8)$$

$$R5 = (1, -4, 6, -4, 1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.. (5.9)$$

$$W5 = (-1, 2, 0, -2, 1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (5.10)$$

By extending the lengths of the kernels, new kernels referred to as Ripple and Wave are obtained. These sets of one-dimensional kernels are combined to generate two-dimensional kernels, which are then used to analyse 2D images. There are 9 two-dimensional kernels (3x3) for kernels of a length of three:

E3E3 E3L3 E3S3

L3E3 L3L3 L3S3

S3E3 S3L3 S3S3

. There are 25 two-dimensional kernels (5x5) for kernels of a length of five:

L5L5 L5E5 L5S5 L5R5 L5W5

E5L5 E5E5 E5S5 E5R5 E5W5

S5L5 S5E5 S5S5 S5R5 S5W5

R5L5 R5E5 R5S5 R5R5 R5W5

W5L5 W5E5 W5S5 W5R5 W5W5

Following are the steps involved in retrieving the above 2-D masks.

a. Convolution is performed between the image $I_{(i,j)}$ and the 2D mask to retrieve the texture image(TI)

   For 5x5 mask, $TI_{E_5E_5} = I_{(i,j)} * E_5E_5 \ldots\ldots\ldots\ldots\ldots\ldots\ldots... (5.11)$

   For 3x3 mask, $TI_{E_3E_3} = I_{(i,j)} * E_3E_3 \ldots\ldots\ldots\ldots\ldots\ldots\ldots... (5.12)$

b. Perform normalization using min-max normalization by passing the image through Texture Energy Measurement (TEM) filters.

For 5x5 mask, $TEM_{ij} = \sum_{u=-5}^{5} \sum_{v=-5}^{5} Normalize(TIi + u, j + v)$……(5.13)

For 3x3 mask, $TEM_{ij} = \sum_{u=-3}^{3} \sum_{v=-3}^{3} Normalize(TIi + u, j + v)$……(5.14)

c. Texture Energy measures (TEMs) denoted as TR are retrieved by blending TEM descriptors.

For 5x5 mask, $TR_{E_5L_5} = \frac{TEME5L5+TEML5E5}{2}$………………(5.15)

For 3x3 mask, $TR_{E_3L_3} = \frac{TEME3L3+TEML3E3}{2}$ ……………….(5.16)

d. Different statistical features are extracted using GLCM. A few of the statistical features are mentioned below:

Contrast is a measurement of the difference in intensity or grey level between the reference pixel and its neighbour. In GLCM, a large contrasting indicates a high difference in intensity

$Contrast = \sum_{i,j=0}^{N-1} P_{ij}(i - j)^2$……………………….(5.17)

where, P_{ij} = Element i,j of the normalized symmetrical GLCM. N = Number of gray levels in the image as specified by Number of levels in under Quantization on the GLCM texture page of the Variable Properties dialog box.

Dissimilarity is the difference in grey level pairs. It's comparable to contrast with the exception that the weights rise in a linear fashion.

$Dissimilarity = \sum_i \sum_j |i - j| GLCM_{(i,j)}$…………………..(5.18)

Homogeneity determines how near the GLCM element arrangement is to the GLCM diagonally.

$Homogeneity = \sum_{i,j=0}^{N-1} \frac{Pij}{1+(i-j)2}$ ……………………..(5.19)

The orderliness of the pixel values in the frame is measured by the Angular Second Moment (ASM).

$ASM = G\ X - 1\ i = 0\ G\ X - 1\ j = 0\ \{P_{(i,j)}\}^2$………………(5.20)

Energy is derived from ASM

$$Entropy = \sum_{i,j=0}^{N-1} \ln(P_{ij})P_{ij} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(5.21)$$

Step 4: Following this, diseased leaf images having 2, 3 and 10 classes for the respective plant types are used to obtain features. Learning is done in several classes employing SVM with "rbf" kernel, and testing is conducted using accuracy, precision and recall.

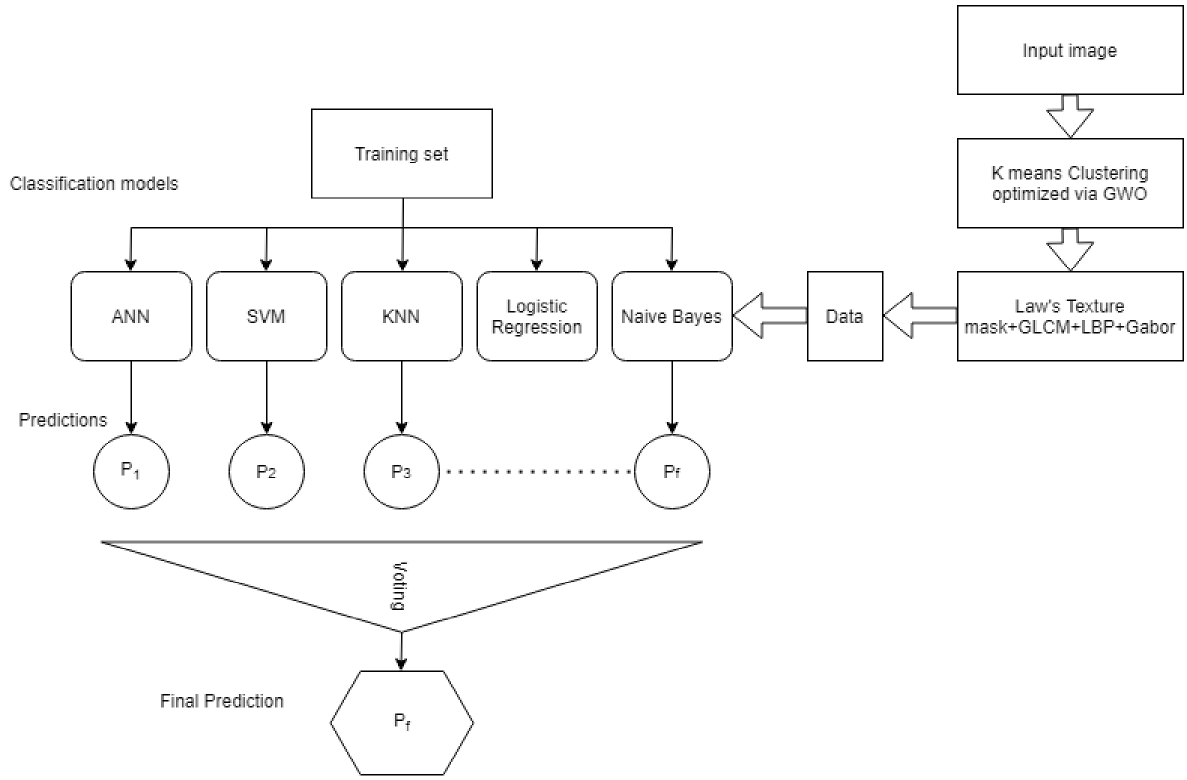


Fig 5.14 Proposed Methodology

5.4 EXPERIMENT AND RESULT ANALYSIS

Since, the best ensemble found is ANN+SVM+KNN+LR+ NB, we have implemented a variety of combinations as per Table 5.1. We have tested 3x3 Law's mask with ensemble classification. Also, the feature reduction technique PCA has been used in the combinations since PCA had given us good results in terms of all performance metrics as compared to LDA (refer to Chapter 4). In Table 5.1 Pca-Rf3 refers to the combination of Law's mask, PCA and RF. The ensemble has been used in conjunction with proposed approach in Chapter 4. Pca-ensemble-3 refers to combination of Law's mask, PCA and the best identified ensemble classifier (ANN, SVM, LR, KNN, NB). Rf-3 refers to the combination of Law's mask and RF. Ensemble-3 refers to the combination of Law's mask and the best identified ensemble classifier (ANN, SVM, LR, KNN, NB).

Table 5.1 Abbreviations for different approaches used

Approach used	**Abbreviation used in results**
Proposed features (3*3 Law's mask) + PCA + RF	Pca-Rf3
Proposed features (3*3 Law's mask) + PCA + (ANN, SVM, Logistic Regression, KNN, Naïve Bayes)	pca-ensemble-3
Proposed features (3*3 Law's mask) + RF	Rf-3
Proposed features (3*3 Law's mask) + (ANN, SVM, Logistic Regression, KNN, Naïve Bayes)	Ensemble-3

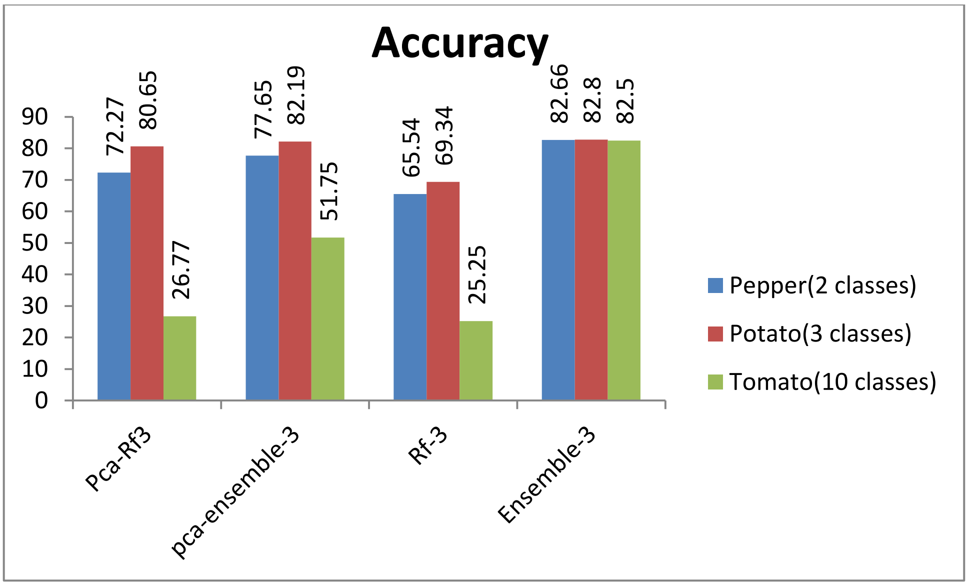


Fig 5.15 Accuracy comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask

Fig 5.15 shows accuracy comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask. Pca-Rf3 has shown accuracy of 72.27% for pepper, 80.65% for potato, and 26.77% for tomato. pca-ensemble-3 has shown an accuracy of 77.65% for pepper, 82.19% for potato, and 51.75% for tomato. Rf-3 has shown an accuracy of 65.54% for pepper, 69.34% for potato, and 25.25% for tomato. Ensemble-3 has shown an accuracy of 82.66% for pepper, 82.80% for potato and 82.50% tomato. The maximum accuracy is yielded by Ensemble-3 and that is 82.66% for pepper plant followed by 82.80% for potato and 82.50% tomato plants.

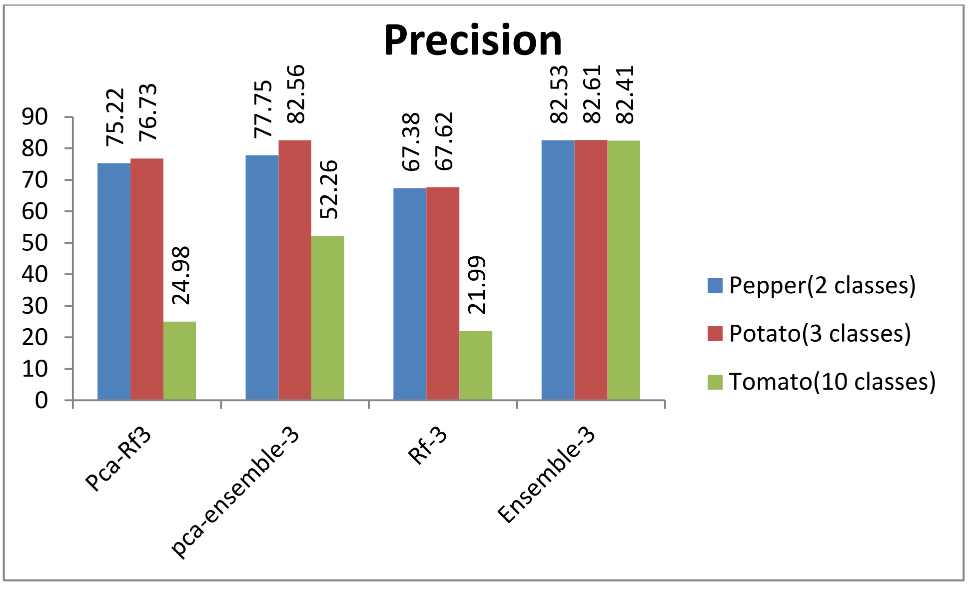


Fig 5.16 Precision comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask

Fig 5.16 shows precision comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask. Pca-Rf3 has shown a precision of 75.22% for pepper, 76.73% for potato, and 24.98% for tomato. pca-ensemble-3 has shown a precision of 77.75% for pepper, 82.56% for potato, and 52.26% for tomato. Rf-3 has shown a precision of 67.38% for pepper, 67.62% for potato, and 21.99% for tomato. Ensemble-3 has shown a precision of 82.53% for pepper, 82.61% for potato and 82.41% for tomato. The maximum precision is yielded by Ensemble-3 and that is

82.61% for potato plant followed by 82.53% for potato plant yielded by Ensemble-3 and 82.41% for tomato plants yielded by Ensemble-3.

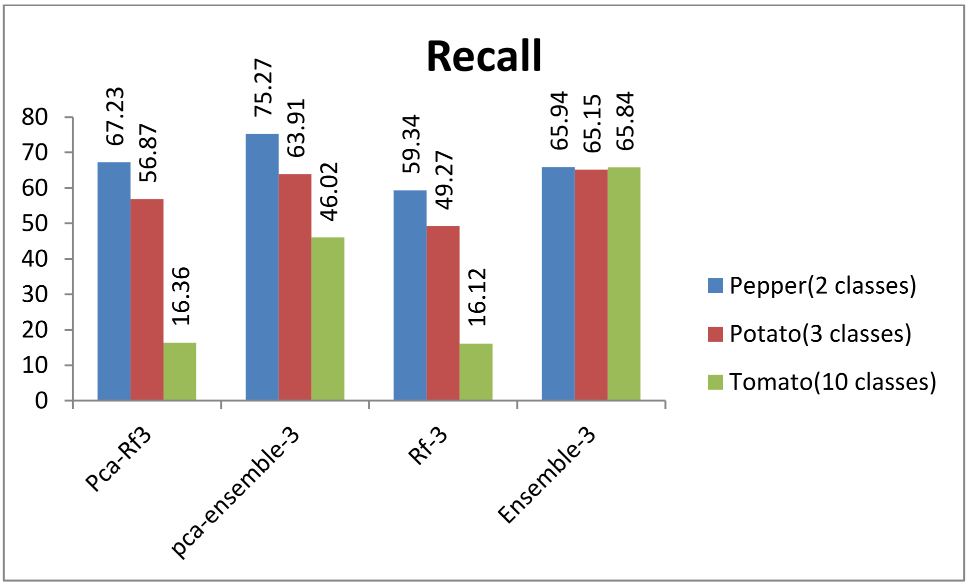


Fig 5.17 Recall comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask

Fig 5.17 shows recall comparison chart for ensemble classification in conjunction with proposed approach using 3x3 Law's mask. Pca-Rf3 has shown a recall of 67.23% for pepper, 56.87% for potato, and 16.36% for tomato. pca-ensemble-3 has shown a recall of 75.27% for pepper, 63.91% for potato, and 46.02% for tomato. Rf-3 has shown a recall of 59.34% for pepper, 49.27% for potato, and 16.12% for tomato. Ensemble-3 has shown a recall of 65.94% for pepper, 65.15% for potato and 65.84% for tomato. The maximum recall is yielded by pca-ensemble-3 and that is 75.27% for pepper plant followed by 65.15% yielded by Ensemble-3 for potato plant and 65.84% yielded by Ensemble-3 for tomato plant.

From Fig 5.15, Fig 5.16 and Fig 5.17, it is evident that 3x3 Law's mask, when used with individual existing classifiers without using any feature reduction techniques perform well as compared to the rest of the combinations. Also it was quite clear from

Chapter 4 that dimensionality reduction reduced the performance of the approach. Also, the ensemble of ANN, KNN, SVM, LR and NB is stronger than Random Forest which is itself an ensemble technique. More number of classifiers in an ensemble approach can improve the results but at the same time they also increase the complexity.

5.5 DETAILS REGARDING TIME COMPLEXITY

Since, we have already proved that Ensemble-3 is the best approach so far in terms of accuracy, precision and recall, but the computation and time complexity increases. More the number of classifiers, more the complexity. Details are mentioned in Table 5.2. Time is mentioned in minutes. It has been observed that Ensemble-3 took 2.2min for pepper, 2.28min for potato and 23.45min for tomato. Rf-3 took 1.59min for pepper, 2.03min for potato, 12.51min for tomato. pca-ensemble-3 took 2.1min for pepper, 1.56min for potato and 22.43min for tomato. Pca-Rf3 took 2.09min for pepper, 1.49min for potato and 17.5min for tomato. An important finding is that Ensemble-3, though performed best but took the maximum execution time thereby increasing the complexity of the algorithm. It has also been observed that all the approaches have consumed comparatively more time for tomato as compared to pepper and potato because the maximum number of classes (10) is in tomato while pepper and potato consist of 2 and 3 classes respectively.

Table 5.2 Time complexity of ensemble approaches

Approach	**Pepper**	**Potato**	**Tomato**
Pca-Rf3	2.09	1.49	17.54
pca-ensemble-3	2.1	1.56	22.43
Rf-3	1.59	2.03	12.51
Ensemble-3	2.2	2.28	23.45

5.6 ENSEMBLE CLASSIFICATION WITH CONVOLUTIONAL NEURAL NETWORK AND RANDOM FOREST

Deep learning has become a popular concept in recent years, and it has proven to be particularly useful in pattern identification. Among the most significant advantages is that it can simplify the feature extraction process. When contrast to other machine learning methods, DL reduces the rate of error, shortens the assessment time, and improves accuracy rate. The fundamental goal in this section is to use a blend of machine learning and deep learning to identify the leaf disease. For feature extraction, CNN is employed, and a hybrid technique of CNN and RF [93] is applied. Since CNN is good for large datasets only but the implemented approach in this section has been specifically used for small-sized dataset and the performance is analyzed.

5.6.1 Model Building

The convolutional layer, the pooling layer, the ReLU correction layer, and the fully-connected layer are the distinct types of layers in CNN. Because the convolutional layer is the first layer, it receives images as input. It calculates the convolution of each image with each of the filters. The parts of the image that we wish to work with are represented by these filters. As a result, we have feature vectors. The pooling layer is in charge of reducing the size of the photos while preserving their important features. It not only reduces the number of parameters and operations in the network, but it also reduces the size of the network. The non-linear function activation function is referred to as ReLu. Finally, there's a fully connected layer that does the categorization of the incoming photos.

A collection of 32 filters are used, followed by max pooling layers to reduce the output's spatial dimensions. A 5x5 kernel size is used for the convolutional layer. A 2x2 Max Pooling is employed later. The ‘relu' activation function is utilized. Finally, for categorization of diseases, RF is integrated with the fully connect layer which ultimately produces the output. We have considered RF because it is a technique that uses ensemble learning and is based on the bagging technique. It grows as many trees as possible on a part of the dataset and then merges the results of all of the trees. As a result, the overfitting issue in decision trees is reduced, as is the variance, which increases accuracy. Fig 5.18 shows the CNN architecture used.

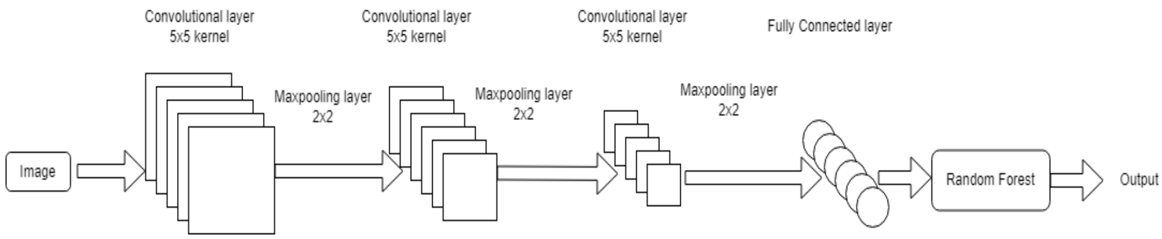


Fig 5.18 CNN architecture

5.6.2 Individual performance of RF and CNN

Here, we have analyzed the individual performance of Random Forest and CNN. As per experimental results in section 5.4, Rf-3 has shown an accuracy of 65.54% for pepper, 69.34% for potato, and 25.25% for tomato. Rf-3 has shown a precision of 67.38% for pepper, 67.62% for potato, and 21.99% for tomato. Rf-3 has shown a recall of 59.34% for pepper, 49.27% for potato, and 16.12% for tomato. It has also been observed that CNN (with model built as per 6.6.1) shows an accuracy of 72% for pepper, 73% for potato and 64% for tomato. Precision shown is 76% for pepper, 74% for potato and 70% for tomato. Recall observed is 72% for pepper, 60% for potato and 65% for tomato. Since RF is itself an ensemble of decision trees, we tested this classifier by integrating it with CNN.

5.6.3 Feature Extraction

The feature extractors are represented by the convolution operation. It captures the important characteristics. The first convolution operation extracts low-level characteristics like edges, corners, and lines. The higher level layers extract high-level characteristics.

5.6.4 Classification

For unidentified plant images, the categorization is employed after the training procedure. The photograph is used as an intake, and the algorithm compares the training and testing photos to determine the disease label.

5.6.5 Comparison

The approach using CNN with RF is compared to Ensemble-3 (refer to section 5.4). Law's texture feature extraction (3x3), Grey Level Co-occurrence matrix (GLCM), Local Binary Pattern (LBP) and ensemble classification of SVM, KNN, Naive Bayes, Logistic regression, and neural network are used in Ensemble-3.

5.6.6 Results and discussion

When contrasted to other machine learning methods, CNN + RF worked satisfactorily. Fig 5.13, 5.14 and 5.15 show the comparison of accuracy, precision and recall respectively.

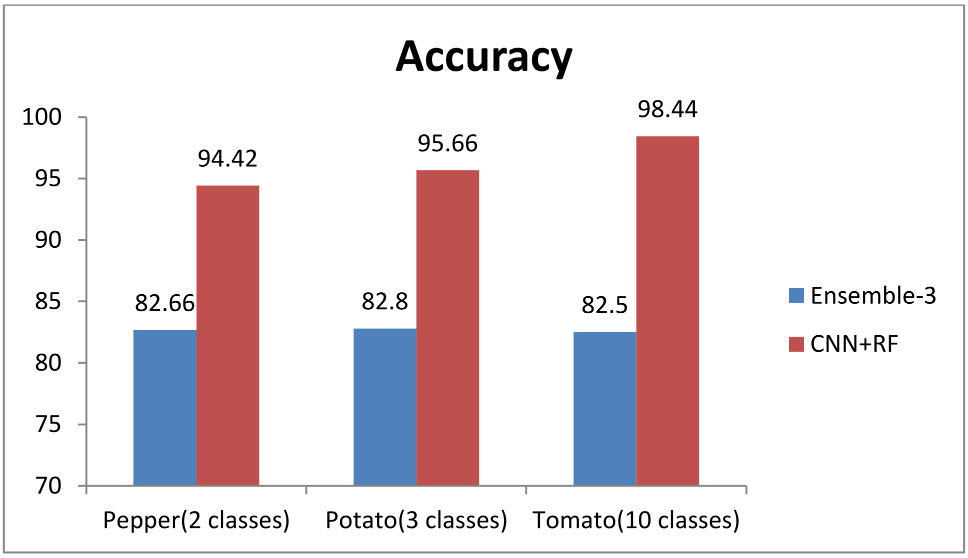


Fig 5.19 Accuracy comparison chart of CNN + RF with Ensemble-3

Fig 5.19 shows the accuracy comparison chart for CNN+RF and Ensemble-3. For bell pepper, the accuracy achieved by CNN+RF is 94.42% as compared to Ensemble-3 that gives 82.66% accuracy. For potato, the accuracy achieved by CNN+RF is 95.66% and by Ensemble-3 is 82.80%. For tomato, the accuracy achieved by CNN+RF is 98.44% as compared to Ensemble-3 that gives 82.50% accuracy.

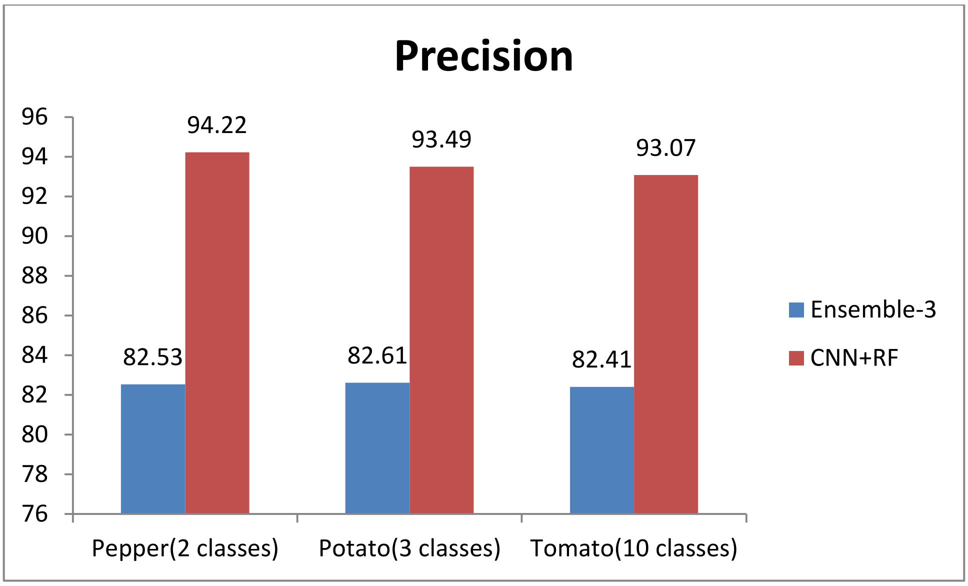


Fig 5.20 Precision comparison chart of CNN + RF with Ensemble-3

Fig 5.20 shows the precision comparison chart for CNN+RF and Ensemble-3. For bell pepper, the precision achieved by CNN+RF is 94.22% as compared to Ensemble-3 that gives 82.53% precision. For potato, the precision achieved by CNN+RF is 93.49% as compared to Ensemble-3 that gives 82.61% precision. For tomato, the precision achieved by CNN+RF is 93.07% as compared to Ensemble-3 that gives 82.41% precision.

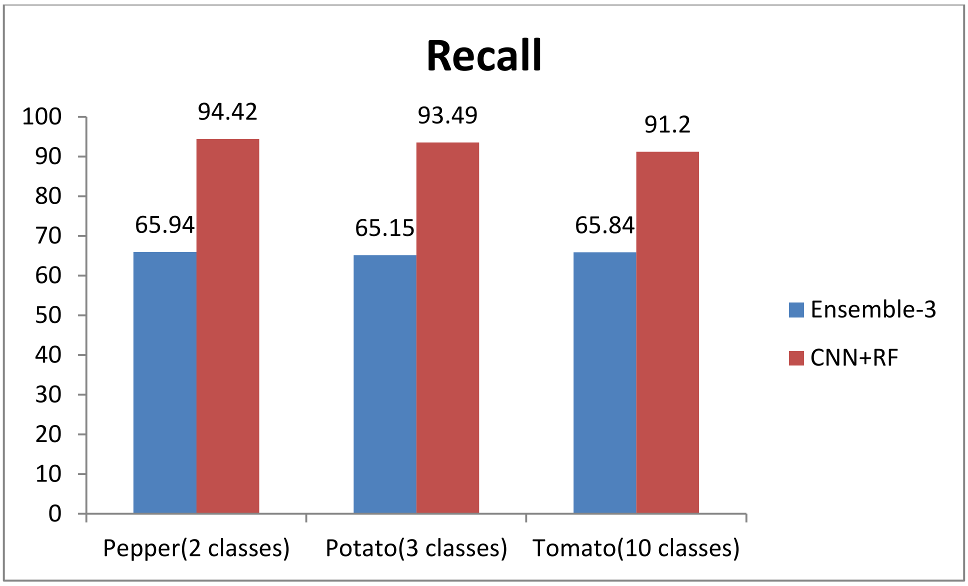


Fig 5.21 Recall comparison chart of CNN + RF with Ensemble-3

Fig 5.21 shows the recall comparison chart for CNN+RF and Ensemble-3. For bell pepper, the recall achieved by CNN+RF is 94.42% as compared to Ensemble-3 that gives 65.94% recall. For potato, the recall achieved by CNN+RF is 93.49% as compared to Ensemble-3 that gives 65.15% recall. For tomato, the recall achieved by CNN+RF is 91.12% as compared to Ensemble-3 that gives 65.84% recall.

It can be concluded that the blend of deep learning and machine learning can not only improve the results but can also reduce the requirement of many traditional machine learning algorithms. We have found that integration of CNN with ensemble based RF classifier can improve the results obtained.

5.7 SUMMARY

The chapter's primary strength is the effective construction of an ensemble classification strategy. Classifiers such as SVM, ANN, KNN, LR, and NB have been employed to create an effective ensemble classifier. The composite classification using several characteristics has been accomplished, and the results have been evaluated. When combined with the proposed approach from Chapter 3, our ensemble classifier produced the best performance in terms of accuracy, precision, and recall. Further, a method is implemented that employs an aggregation of CNN model and RF which also shows that CNN can work efficiently with small-sized datasets.

CHAPTER 6

LEAF DISEASE DETECTION USING ENSEMBLE CLASSIFIER, GABOR FILTER AND SIFT

This chapter focuses on using ensemble classification in conjunction with hybrid Law's mask, LBP, GLCM, SIFT, and Gabor features to boost classification results.

6.1 EXPERIMENT AND RESULT ANALYSIS

This chapter is an extension of Chapter 5 where we had already concluded that Ensemble-3 (Proposed features (3*3 Law's mask) + (ANN, SVM, Logistic Regression, KNN, Naïve Bayes) performed pretty well in all the three classes. Though we also proposed an ensemble of CNN and RF which provided very good results in terms of accuracy, precision and recall, our focus in this thesis is mainly on machine learning approach. So here in this chapter we have further improved the proposed approach (Ensemble-3) using various other feature extraction approaches. This time along with the already used features (Law's mask, LBP and GLCM), Gabor [52] and SIFT [105] have also been utilized in order to analyse the results. Gabor filter examines whether the image contains any specific frequency content in specified direction in a localized area surrounding the analysis point or region. SIFT is used to describe local features in an image that provides qualitative information in the form of descriptors. The results of the performance assessment are shown in Fig 6.1, 6.2, and 6.3.

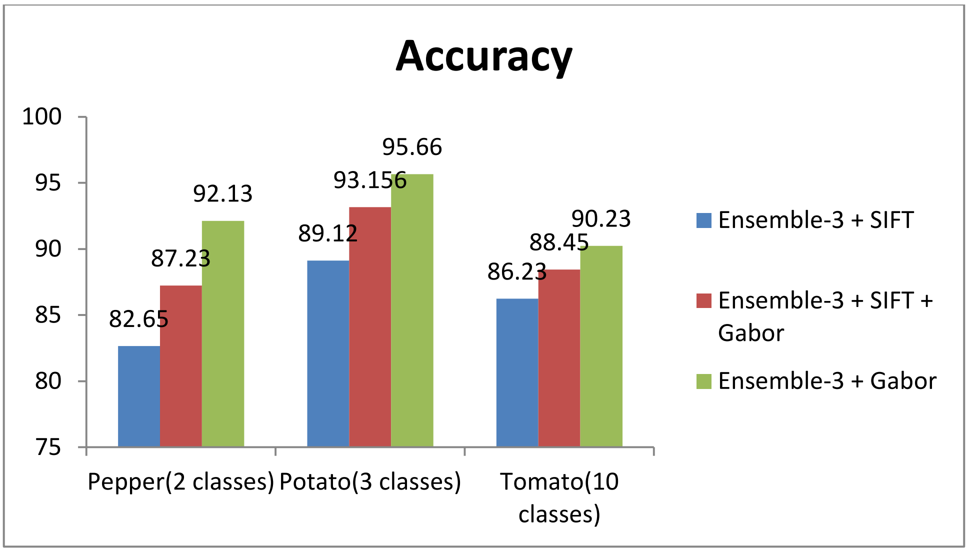


Fig 6.1 Accuracy comparison chart of Ensemble-3 in conjunction with Gabor and SIFT

Fig 6.1 shows the accuracy results of combinations of the proposed approach Ensemble-3 with Gabor and SIFT. It has been observed that Ensemble-3 + Gabor has shown the best accuracy – 92.13% for pepper, 95.66% for potato and 9023% for tomato.

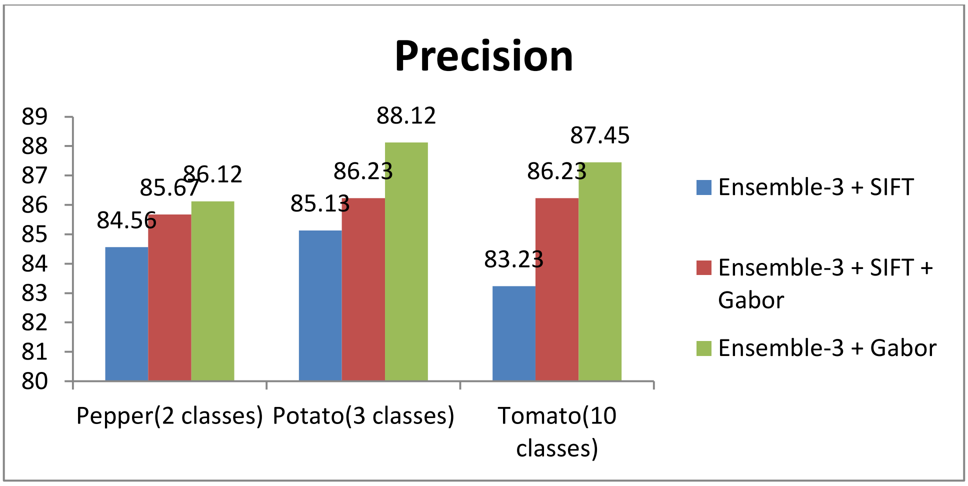


Fig 6.2 Precision comparison chart of Ensemble-3 in conjunction with Gabor and SIFT

Fig 6.2 shows the precision results of combinations of the proposed approach Ensemble-3 with Gabor and SIFT. It has been observed that Ensemble-3 + Gabor has shown the best precision – 86.12% for pepper, 88.12% for potato and 87.45% for tomato

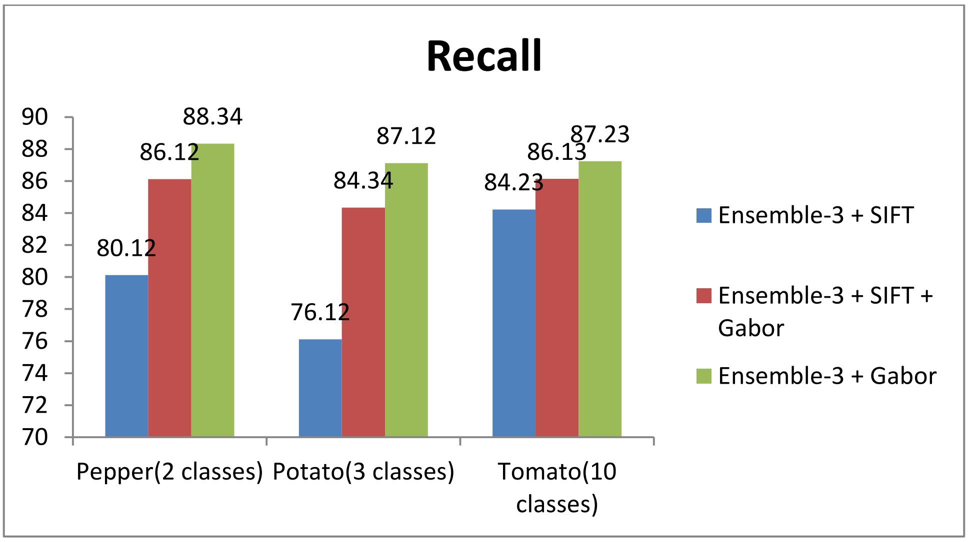


Fig 6.3 Recall comparison chart of Ensemble-3 in conjunction with Gabor and SIFT

Fig 6.3 shows the recall results of combinations of the proposed approach Ensemble-3 with Gabor and SIFT. It has been observed that Ensemble-3 + Gabor has shown the best recall – 88.34% for pepper, 87.12% for potato and 87.23% for tomato

It can be clearly concluded that Gabor features when integrated with the proposed approach gives us very satisfying results in terms of accuracy, precision and recall while on the other hand SIFT features also improved the results but they were not better than when approach was combined with Gabor features. We have further computed a confusion matrix for all the three classes individually.

Table 6.1 Confusion matrix for pepper plant type

	Predicted Values		
		Bacterial Spot	**Healthy**
Actual Values	**Bacterial Spot**	299	35
	Healthy	24	385

Table 6.1 shows the confusion matrix for pepper class. The False Negative Rate(FNR) is 0.1047. The False Positive Rate(FPR) is 0.0586.

Table 6.2 Confusion matrix for potato plant type

	Predicted Values			
		Early Blight	**Healthy**	**Late Blight**
	Early Blight	336	11	23
Actual Values	**Healthy**	6	39	24
	Late Blight	24	42	131

Table 6.2 shows the confusion matrix for potato class. The False Negative Rate(FNR) is 0.0551. The False Positive Rate(FPR) is 0.108.

Table 6.3 Confusion matrix for tomato plant type

	Predicted Values										
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	0	913	3	6	5	21	20	67	72	11	10
	1	25	33	38	47	1	51	12	12	0	57
	2	12	1	360	20	9	61	2	6	1	38
	3	18	4	35	269	8	80	29	3	0	92
Actual	**4**	16	2	32	14	83	64	5	4	1	29

Values	**5**	12	1	18	14	3	403	2	16	11	6
	6	20	2	46	29	3	83	143	13	0	106
	7	25	2	50	24	0	176	25	253	8	19
	8	23	0	5	2	0	69	0	20	12	2
	9	20	6	15	12	3	9	45	4	0	206

Table 6.3 shows the confusion matrix for tomato class. For confusion matrix the classes are represented as numbers as:

0 Bacterial Spot

1 Early Blight

2 Healthy

3 Late Blight

4 Leaf mod

5 Septoria leaf spot

6 Spider mites

7 Target spot

8 Mosaic virus

9 Yellow leaf curl virus

The False Negative Rate(FNR) is 0.1577. The False Positive Rate(FPR) is 0.0599.

6.2 COMPARATIVE STUDY

The final proposed approach (Ensemble-3 + Gabor) has been compared with approaches proposed in the recent article which has worked on the same dataset and plants. Our approach has been compared with [104] and [105]. The existing approach [104] has utilized color transformation. Following the extraction of feature sets using a bag of visual words, Fisher vectors, and handmade features, classification using logistic regression, multilayer perceptron model, and support vector machine is performed. The suggested algorithm's contribution was to optimize the retrieved information from the available resources for improved results without adding any more complexity. PlantVillage datasets of apple, bell pepper, cherry, corn, grape, potato, and tomato are used to evaluate the proposal's performance. In [105], the process of county expansion along with Fisher vectors has been performed. The suggested technique first uses the color properties of the leaf image to localize the leaf region, followed by a mixture model based county expansion for leaf localization. The discriminatory qualities of the leaf images are used to classify the photographs. The diseased images' distinguishing features reveal a variety of patterns in the leaf region. The Fisher vector was used to exploit the features' discriminable quality in terms of different orders of differentiation of Gaussian distributions. Using the PlantVillage databases of common pepper, root vegetable such as potato, and tomato leaf photos, the performance of the suggested system is evaluated using a multi-layer perceptron and SVM. Table 6.4 presents the comparison of accuracy of proposed approach with existing approaches.

Table 6.4 Accuracy comparison of proposed approach with approaches used in previous publications

Plant	**Proposed Approach**	**Bag of visual words + Fisher vectors [104]**	**County Expansion + Fisher vector + MLP Classifier [105]**	**County Expansion + Fisher vector + SVM Classifier [105]**
Pepper	92.13	**96.1**	92.8	95.5
Potato	**95.66**	91.9	92.9	94.4
Tomato	90.23	88.2	89.2	**91.8**

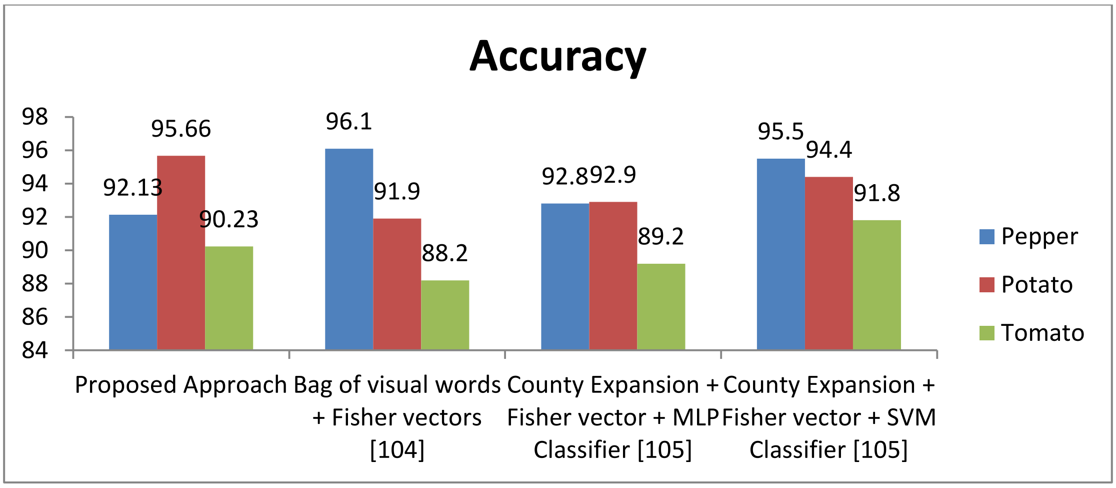


Fig 6.4 Comparison of accuracy of proposed approach with existing approach

The graph for classification accuracy comparison of the proposed approach with state-of-the-art approaches is depicted in Fig 6.4. The accuracy comparison chart (which is depicted above) reveals the following observations. For potato, our proposed approach has shown the best result achieving a maximum accuracy of 95.66% followed by 94.4% achieved by County Expansion + Fisher vector + SVM Classifier [104], 92.9% achieved by County Expansion + Fisher vector + MLP Classifier [105] and 91.9% achieved by Bag of visual words + Fisher vectors [104]. For bell pepper, Bag of visual words + Fisher [104] vectors outperforms our approach and other approaches with an accuracy of 95.5%. For tomato, the proposed approach has revealed the second highest accuracy of 90.23%. Highest accuracy for tomato is 91.8% and has been achieved by County Expansion + Fisher vector + SVM [105] followed by the proposed approach achieving an accuracy of 90.23%. 89.2% accuracy is achieved by County Expansion + Fisher vector + MLP Classifier [105] and 88.2% accuracy achieved by Bag of visual words + Fisher vectors [104].

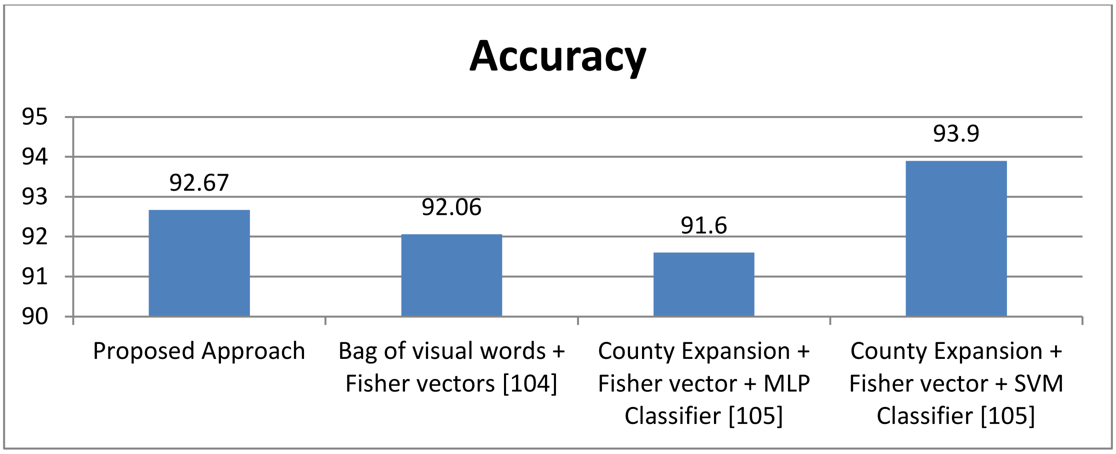


Fig 6.5 Comparison of average accuracy between proposed approach and existing approach

The proposed approach (Fig 6.5) has revealed an average accuracy of 92.93%. The maximum accuracy is achieved by County Expansion + Fisher vector + SVM Classifier (93.9%) followed by the proposed approach (92.67%), Bag of visual words + Fisher vectors (92.06%) and County Expansion + Fisher vector + MLP Classifier (91.6%)

6.3 SUMMARY

Gabor features, and SIFT features have been tested with approach proposed in Chapter 5. Out of all combinations, approach using ensemble learning and Gabor filter produced the best performance in terms of accuracy, precision, and recall. Bell pepper (two categories) had 92.13% accuracy, potato (three categories) had 95.66% accuracy, and tomato had 90.23% accuracy (10 categories). We have calculated the percentage in improvement of this final proposed approach from Ensemble-3 (proposed in Chapter 5). For bell pepper, the improvement is 9.477% in accuracy, 3.59% in precision, 22.4% in recall. For potato, the improvement is 12.86% in accuracy, 5.51% in precision and 21.97% in recall. For tomato, the improvement is 7.73% in accuracy, 5.04% in precision and 21.39% in recall. We have also calculated the percentage in improvement in average accuracy from the existing approaches as mentioned in Section 6.2. The average accuracy improvement from Bag of visual words + Fisher vectors is found to be 0.61% while from County Expansion + Fisher vector + MLP Classifier, the average accuracy improvement found to be is 1.07%.

CHAPTER 7

PROPOSED SEMI-AUTOMATED ARCHITECTURE FOR LEAF DISEASE DETECTION

The use of automated image processing to detect sickness in leaves minimizes the need for farmers to ensure the safety of farm produce. In this chapter, a semi-automated approach is proposed that predicts disease based on the user's input (a diseased leaf image). The user must choose a plant type from the available selections and then upload the image. The uploaded image is tested, and the correct disease is determined. The system comprises the user and the admin interface and the specific functionalities are mentioned along with the architecture.

7.1 PROPOSED SYSTEM ARCHITECTURE

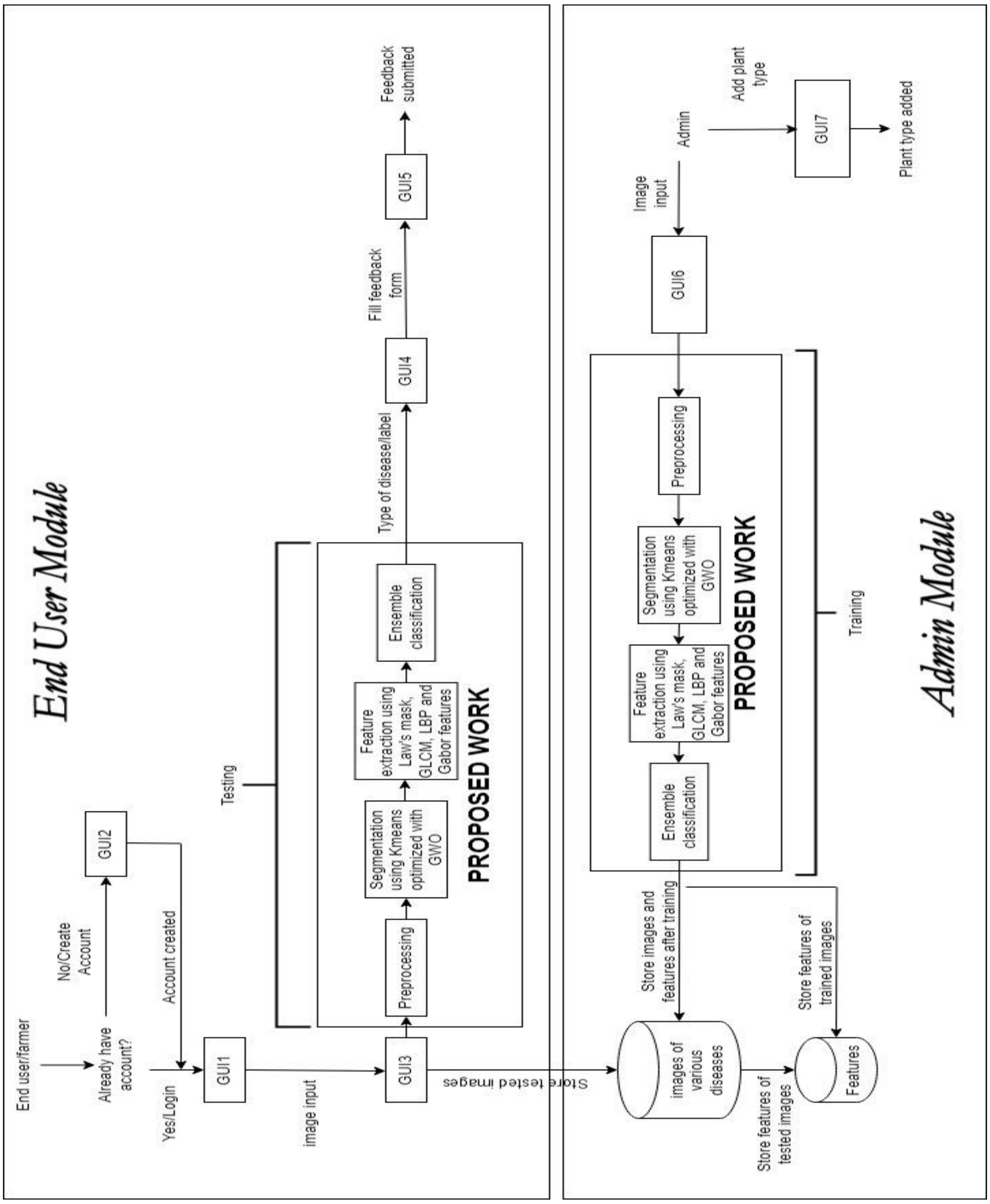


Fig 7.1 Proposed system architecture for plant leaf disease detection

Fig 7.1 shows the overall proposed system architecture for leaf disease detection. The proposed system comprises two modules – the end user module and the admin module. The end user (a farmer) is required to login through GUI1. If the end user is not having an account on the portal, then he needs to create one through GUI2. The end user uploads the image with the help of GUI3. The required processing takes place for this testing. While testing of the image takes place, the image along with its extracted features is stored in the database. The testing approach comprises the proposed approach. The proposed approach consists of preprocessing, segmentation using K-means clustering optimized with GWO, feature extraction using Law's mask, GCM, LBP and Gabor filter, and a classifier that is an ensemble of various classifiers. After processing through the proposed work, the type of disease is predicted in the form of labels. The analysis report will be available on GUI4. After this, the end user has an option to fill the feedback form. The admin module has been given certain rights such as training new images, adding new plant types and deletion of old data including images and features. The new plant type can be added through GUI7 while GUI6 is used to upload new image and train them. For training, the images pass through the proposed work as mentioned. The trained images and their corresponding features are stored in the database.

7.2 USER INTERFACE FOR PLANT LEAF DISEASE DETECTION

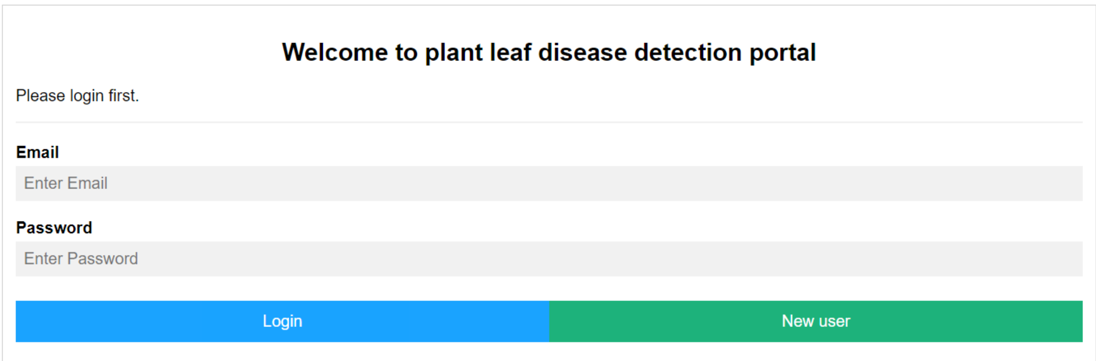


Fig 7.2 End user Login interface (GUI1)

Fig 7.2 shows the user interface for the existing end user. The interface requires the existing end user to fill email ID and password to login into the plant leaf disease detection portal. The blue button logs the user into the portal while the green button is for a new user.

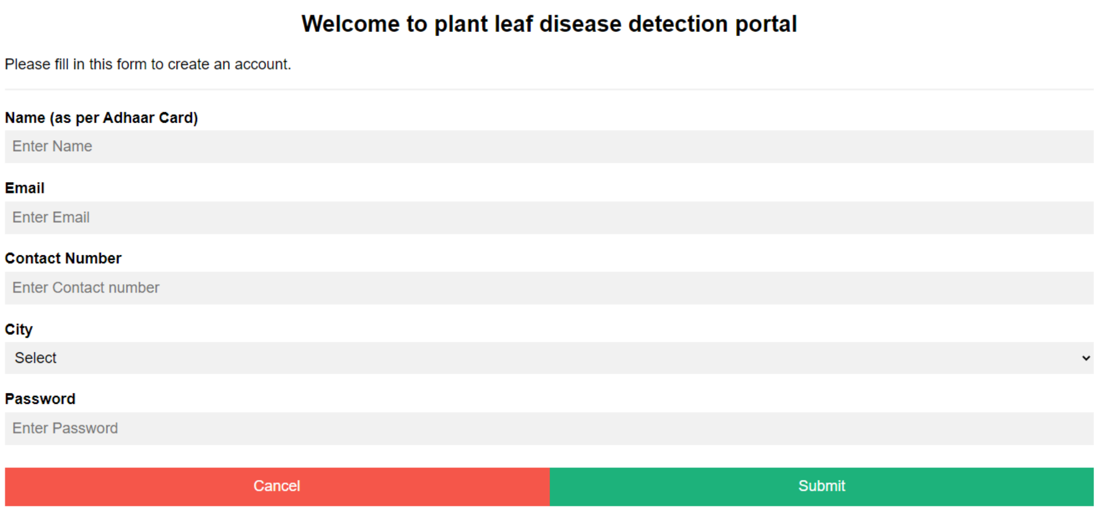


Fig 7.3 End user Sign up interface (GUI2)

Fig 7.3 shows the interface for signup by a new user. In GUI1, if the user clicks on New User button, the user is redirected to GUI2 for account creation. The new user is required to fill his/her name as per Adhaar card, email ID, contact number, city and password. The green submit button creates a new user while the cancel button is for cancellation of account creation.

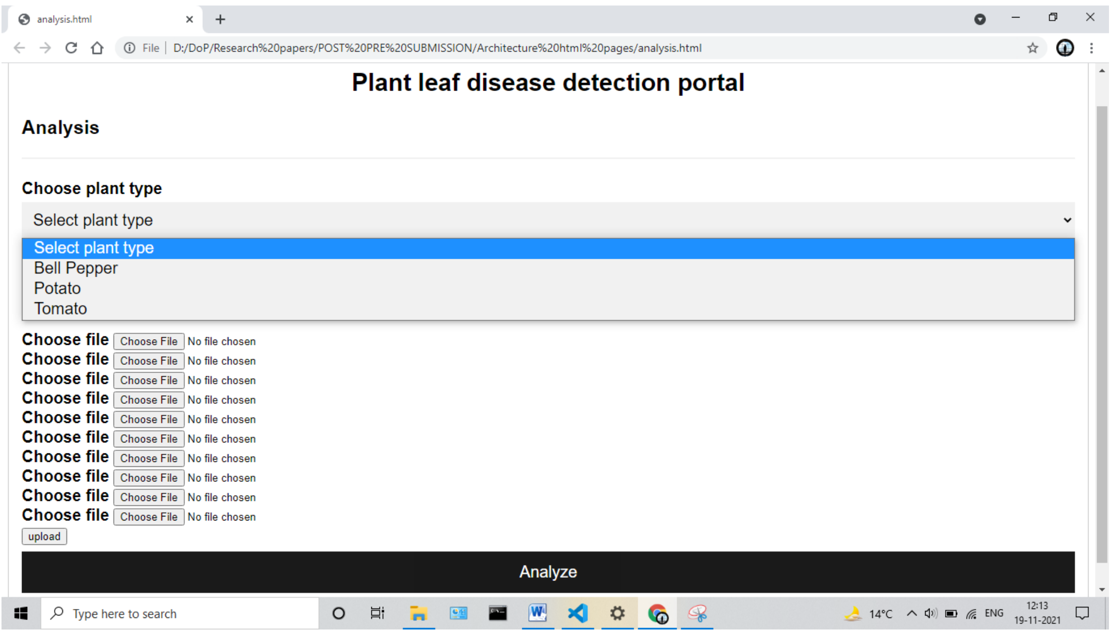
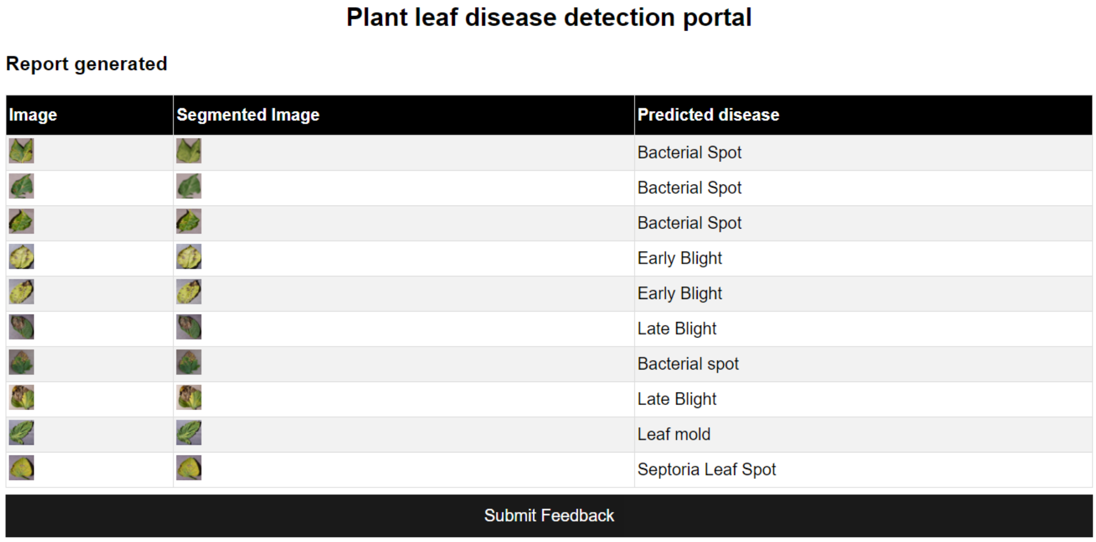


Fig 7.4 Choose plant type, browse image and analyze the result (GUI3)

After the user logs in, user is redirected to GUI3 as in Fig 7.4 where the user is required to select the plant type and upload as many images as the user wants corresponding to that particular plant type. The user later clicks on the Analyze button to get the analysis report.


Plant leaf disease detection portal

Report generated

Image	Segmented Image	Predicted disease
		Bacterial Spot
		Bacterial Spot
		Bacterial Spot
		Early Blight
		Early Blight
		Late Blight
		Bacterial spot
		Late Blight
		Leaf mold
		Septoria Leaf Spot

Submit Feedback

Fig 7.5 Segmented images and analysis report generated for uploaded images (GUI4)

Fig 7.5 shows the analysis report (GUI4) after the user clicks on "Analyze" button in GUI3. The analysis report consists of a table in which the first column shows the original uploaded image, second column shows the segmented image after processing and third column shows the predicted disease.

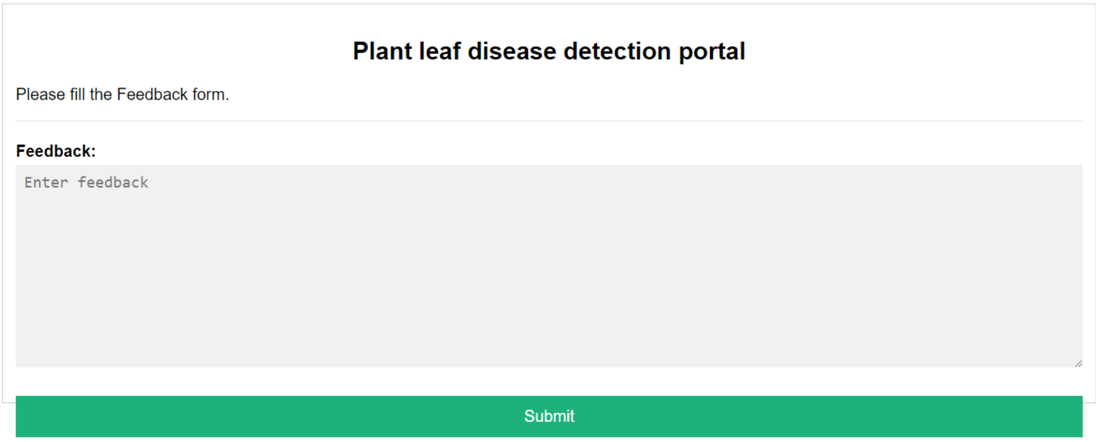


Fig 7.6 Feedback interface (GUI5)

From GUI4, the user is redirected to GUI5 after clicking on Submit Feedback button, as in Fig 7.6. The user is required to fill the feedback and then click on Submit.

7.3 ADMIN INTERFACE FOR PLANT LEAF DISEASE DETECTION

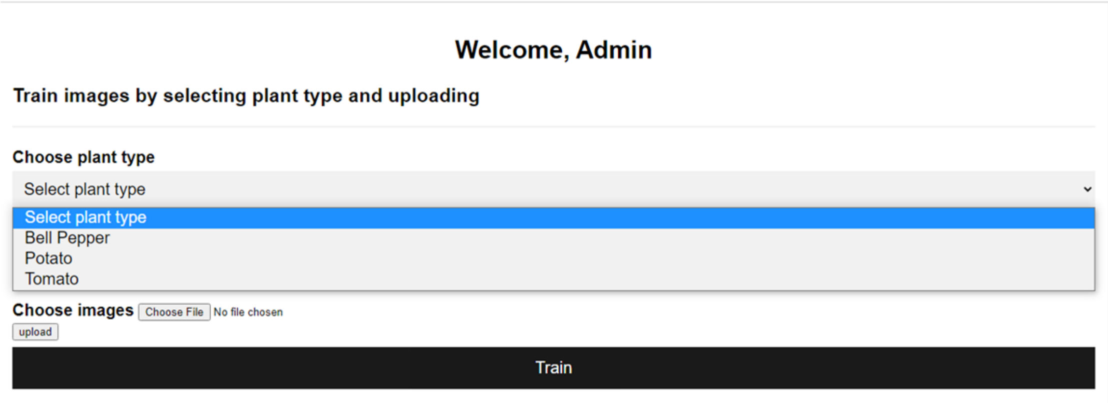


Fig 7.7 Choose plant type, upload image and train at a click (GUI6)

GUI6 (Fig 7.7) lets the admin to add and train new images into the system with the click of a button after choosing the plant type from the dropdown menu.

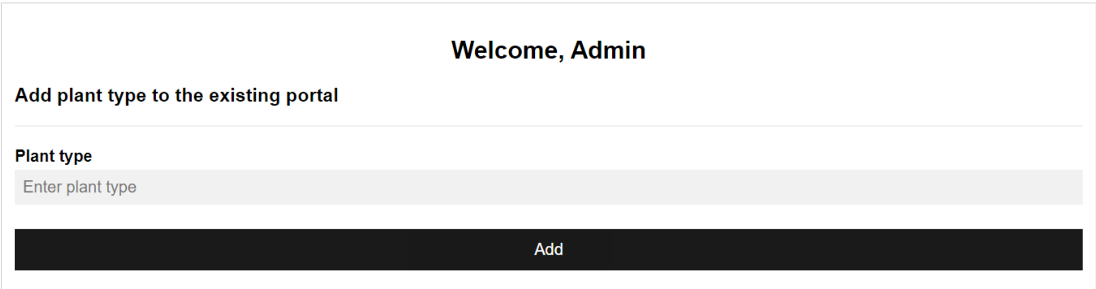


Fig 7.8 Add plant type to the existing portal (GUI7)

GUI7 (Fig 7.8) allows the admin to add the plant type by entering the new plant type in the given text box field and then clicking on Add button.

7.4 SUMMARY

A system architecture of the proposed semi-automated system has been presented. The graphical user interfaces have also been proposed in the system. The system is intended to be implemented as a future work.

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

This chapter summarizes the chapter-wise results. Conclusion is presented along with the future research directions.

8.1 SUMMARY OF RESULTS

This section provides a brief summary of chapters suggesting the proposed approaches.

From Chapter 3, we were able to investigate the best hybrid as Clustering segmentation optimized with GWO + Law's texture mask (3x3) + LBP + GLCM + SVM based on accuracy, precision and recall. Average percentage in improvement for the proposed approach (Clustering segmentation optimized with GWO + Law's texture mask (3x3) + LBP + GLCM + SVM) from other combinations (in Chapter 3) was calculated. The average improvement is 4.89% in accuracy, 9.03% in precision and 6.11% in recall for bell pepper. The average improvement is 4.25% in accuracy, 3.69% in precision and 10.52% in recall for potato. The average improvement is 9.68% in accuracy, 10.52% in precision and 9.43% in recall for tomato.

In Chapter 4, the approach has been tested with PCA and LDA from which we concluded that PCA performs better as compared to LDA but the results have been otherwise found to reduce (in comparison with proposed approach in Chapter 3) in terms of performance metrics. PCA has shown 8.12% reduction in accuracy for pepper, 5.16% reduction in accuracy for potato and 5.51% reduction in accuracy for tomato. It has shown 8.32% reduction in precision for pepper, 4.85% reduction in precision for potato and 6.06% reduction in precision for tomato. Also, it has shown 8.71% reduction in recall for pepper, 9.84% reduction in recall for potato and 3.65% reduction in recall for tomato.

We extended our work further in Chapter 5 by investigating the best ensemble classifier to be used with the proposed approach. We concluded that based on the performance metrics the ensemble of ANN+KNN+SVM+LR+NB provided us the best results based on accuracy, precision and recall. We have also used a hybrid of machine learning and deep learning in the same chapter. Hence, we also tested an ensemble of CNN and RF from which we concluded that CNN can work well with large datasets as well. This proposed ensemble approach (ANN+KNN+SVM+LR+NB) or Ensemble -3 was further improved using Gabor filter in Chapter 6

The final approach was proposed in Chapter 6 where we found the hybrid of Ensemble-3 and Gabor provided the best results. The proposed approach was also tested with existing approaches. Our approach has shown robust results in terms of accuracy precision and recall. The percentage in improvement of hybrid of Ensemble-3 and Gabor from Ensemble-3 (proposed in Chapter 5) was found to be 9.47% in accuracy, 3.59% in precision and 22.4% in recall for bell pepper. 12.86% improvement in accuracy, 5.51% improvement in precision and 21.97% improvement in recall was found for potato. 7.73% improvement in accuracy, 5.04% improvement in precision and 21.39% improvement in recall was found for tomato. The average accuracy improvement from Bag of visual words + Fisher vectors is found to be 0.61% while from County Expansion + Fisher vector + MLP Classifier, the average accuracy improvement found to be is 1.07%.

8.2 CONCLUSION

The most significant implication of the research is the focus on development of a semi-automated system for plant leaf disease detection which will help the farmers in detecting leaf diseases timely and efficiently without an expert's help. A number of hybrid approaches have been tested in our research indicating the progress of the proposed work and finally its comparison with the existing approaches has been presented. For this system, the proposed approach using ensemble classification in conjunction with optimized segmentation and hybrid feature extraction techniques will be used. The proposed approach is 92.67% efficient

8.3 FUTURE SCOPE

The semi-automated system for plant leaf disease detection will be implemented and hosted as a web application to help the farmers. The system will consist of two modules – End user module and the Admin module. This system will use the proposed architecture as described in Chapter 7. It has a potential in terms of further improvement such as its maintenance and using latest techniques and algorithms to update the system.

LIST OF PUBLICATIONS

- N. Kaur, V. Devendran, and S. Verma, “Detection of plant leaf diseases by applying image processing schemes,” *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 9. pp. 3728–3734, 2019, doi: 10.1166/jctn.2019.8241.
- N. Kaur, V. Devendran, and S. Verma, “Plant leaf disease identification supported by image segmentation, feature extraction and ensemble classification,” *International Journal of Control and Automation*, vol. 12, no. 4, pp. 121–133, 2019.
- N. Kaur and V. Devendran, “Novel plant leaf disease detection based on optimize segmentation and Law mask feature extraction with SVM classifier,” *Materials Today: Proceedings*, 2020, doi: 10.1016/j.matpr.2020.10.901.
- N. Kaur and V. Devendran, “Crop plant leaf disease prediction with law mask features using PCA and LDA with SVM,” *International Conference on Challenges and Opportunities for Innovation in New Normal Scenario KSR Institute for Engineering and Technology, Tiruchengode, Namakkal (Dt), Tamil Naduno. April*, pp. 20–30, 2021.
- N. Kaur and V. Devendran, “Plant leaf disease detection using ensemble classification and feature extraction”, *Turkish Journal of Computer and Mathematics Education*,” vol. 12, no. 11, pp. 2339–2352, 2021.
- Navneet Kaur, Dr. V. Devendran, “Plant leaf disease detection using CNN and ensemble classification,” *Design Engineering (Toronto)*, pp. 2372 - 2380, Jun. 2021.
- Navneet Kaur, Dr. V. Devendran, “Plant leaf disease detection using CNN and ensemble classification”, *2nd National Conference of Education, Research and Innovation (ERI) Osmania University Centre for International Program, Osmania University Campus, Hyderabad (India)*, Jun. 2021.
- N. Kaur and V. Devendran, "Ensemble classification and feature extraction based plant leaf disease recognition," *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pp. 1-4, 2021. doi: 10.1109/ICRITO51393.2021.9596456.

- N. Kaur and V. Devendran, "Plant leaf disease diagnostic system built on feature extraction and ensemble classification," *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pp. 1-3, 2021. doi: 10.1109/ICRITO51393.2021.9596070.
- N. Kaur, V. Devendran, and S. Verma, "Crop leaf disease classification identification based on ensemble," *Proceedings of Algorithms, Computing and Mathematics Conference 2021 (ACM 2021)*, vol. 3010, pp. 110–118, 2021.

BIBLIOGRAPHY

[1]. E. Hwa and T. W. Bank, “The Contribution of Agriculture to Economic Growth: Some Empirical Evidence,” *World Development*, vol. 16, no. 11, 1986, doi: 10.1016/0305-750X(88)90208-2

[2]. E. Kennedy, “Commercialization of Subsistence Agriculture: Income and Nutritional Effects in Developing Countries”, 1986.

[3]. R. E. Gaunt, “The relationship between plant disease,” *Annual Review of Phytopathology*, vol. 33, pp. 119-114, 1995, doi: 10.1146/annurev.py.33.090195.001003

[4]. M. Rachidi, A. Marchadier, C. Gadois, E. Lespessailles, C. Chappard, and C. L. Benhamou, “Laws’ masks descriptors applied to bone texture analysis: An innovative and discriminant tool in osteoporosis,” *Skeletal Radiology*, vol. 37, no. 6, pp. 541–548, 2008, doi: 10.1007/s00256-008-0463-2.

[5]. M. Heikkilä, M. Pietikäinen, and C. Schmid, “Description of interest regions with local binary patterns,” *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009, doi: 10.1016/j.patcog.2008.08.014.

[6]. A. R. Backes, W. N. Gonçalves, A. S. Martinez, and O. M. Bruno, “Texture analysis and classification using deterministic tourist walk,” *Pattern Recognition*, vol. 43, no. 3, pp. 685–694, 2010, doi: 10.1016/j.patcog.2009.07.017.

[7]. S. Sankaran, A. Mishra, R. Ehsani, and C. Davis, “A review of advanced techniques for detecting plant diseases,” *Computers and Electronics in Agriculture*, vol. 72, no. 1, pp. 1–13, 2010, doi: 10.1016/j.compag.2010.02.007.

[8]. Wheeler and M. Kay, “Food crop production, water and climate change in the developing world,” *Outlook on Agriculture*, vol. 39, no. 4, pp. 239–243, 2010, doi: 10.5367/oa.2010.0017.

[9]. M. Zhang and Q. Meng, “Automatic citrus canker detection from leaf images captured in field,” *Pattern Recognition Letters*, vol. 32, no. 15, pp. 2036–2046, 2011, doi: 10.1016/j.patrec.2011.08.003.

[10]. A. J. Afifi and W. M. Ashour, “Image Retrieval Based on Content Using Color Feature,” *International Scholarly Research Notices*, vol. 2012, pp. 1–11, Mar. 2012, doi: 10.5402/2012/248285.

[11]. J. K. Kamarainen, “Gabor features in image analysis,” *2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 13–14, 2012, doi: 10.1109/IPTA.2012.6469502.

[12]. S. Panda, S. Sahu, P. Jena, and S. Chattopadhyay, “Comparing Fuzzy-C Means and K-Means Clustering Techniques: A Comprehensive Study,” *Advances in Computer Science, Engineering & Applications*, pp. 451–460, 2012, doi: 10.1007/978-3-642-30157-5_45

[13]. S. Prasad, P. Kumar, R. Hazra, and A. Kumar, “Plant leaf disease detection using Gabor wavelet transform,” *International Conference on Swarm, Evolutionary, and Memetic Computing*, vol. 7677 LNCS, pp. 372–379, 2012, doi: 10.1007/978-3-642-35380-2_44.

[14]. D. Majumdar, A. Ghosh, D. K. Kole, A. Chakraborty and D. D Majumder, “Application of Fuzzy C-Means Clustering Method to Classify Wheat Leaf Images Based on the Presence of Rust Disease”, *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014*, pp. 277–284, 2014, doi: 10.1007/978-3-319-11933-5.

[15]. R. Zhou, S. Kaneko, F. Tanaka, M. Kayamori, and M. Shimizu, “Disease detection of Cercospora Leaf Spot in sugar beet by robust template matching,” *Computers and Electronics in Agriculture*, vol. 108, pp. 58–70, 2014, doi: 10.1016/j.compag.2014.07.004.

[16]. S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[17]. A. S. Setiawan, Elysia, J. Wesley, and Y. Purnama, “Mammogram Classification using Law’s Texture Energy Measure and Neural Networks,” *Procedia Computer Science*, vol. 59, no. Iccsci, pp. 92–97, 2015, doi: 10.1016/j.procs.2015.07.341.

[18]. K. Sumithra, S. Buvana, and R. Somasundaram, “A Survey on Various Types of Image Processing Technique,” *International Journal of Engineering Research and*, vol. V4, no. 03. 2015, doi: 10.17577/ijertv4is030552.

[19]. S. Shrivastava, S. K. Singh, and D. S. Hooda, “Color sensing and image processing-based automatic soybean plant foliar disease severity detection and

estimation," *Multimedia Tools and Applications*, vol. 74, no. 24, pp. 11467–11484, 2015, doi: 10.1007/s11042-014-2239-0.

[20]. S. W. Zhang, Y. J. Shang, and L. Wang, "Plant disease recognition based on plant leaf image," *Journal of Animal and Plant Sciences*, vol. 25, no. 3, pp. 42–45, 2015.

[21]. Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic Study of Automated Diagnosis of Viral Plant Diseases Using Convolutional Neural Networks", *International Symposium on Visual Computing*, vol. 9475, no. 1, pp. 842–850, 2015, doi: 10.1007/978-3-319-27863-6.

[22]. W. Kazmi, F. J. Garcia-Ruiz, J. Nielsen, J. Rasmussen, and H. Jørgen Andersen, "Detecting creeping thistle in sugar beet fields using vegetation indices," *Computers and Electronics in Agriculture*, vol. 112, pp. 10–19, 2015, doi: 10.1016/j.compag.2015.01.008.

[23]. W. Yang, S. Wang, X. Zhao, J. Zhang, and J. Feng, "Greenness identification based on HSV decision tree," *Information Processing in Agriculture*, vol. 2, no. 3–4, pp. 149–160, 2015, doi: 10.1016/j.inpa.2015.07.003.

[24]. A. K. Dey, M. Sharma, and M. R. Meshram, "Image Processing Based Leaf Rot Disease, Detection of Betel Vine (Piper BetleL.)," *Procedia Computer Science*, vol. 85, no. Cms, pp. 748–754, 2016, doi: 10.1016/j.procs.2016.05.262.

[25]. B. VijayaLakshmi and V. Mohan, "Kernel-based PSO and FRVM: An automatic plant leaf type detection using texture, shape, and color features," *Computers and Electronics in Agriculture*, vol. 125, pp. 99–112, 2016, doi: 10.1016/j.compag.2016.04.033.

[26]. J. D. Pujari, R. Yakkundimath, and A. S. Byadgi, "SVM and ANN Based Classification of Plant Diseases Using Feature Reduction Technique," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 3, no. 7, p. 6, 2016, doi: 10.9781/ijimai.2016.371.

[27]. J. G. A. Barbedo, "A novel algorithm for semi-automatic segmentation of plant leaf disease symptoms using digital image processing," *Tropical Plant Pathology*, vol. 41, no. 4, pp. 210–224, 2016, doi: 10.1007/s40858-016-0090-8.

[28]. M. P. Pawar, T. Varsha, and P. Pravin, "Cucumber Disease Detection Using Artificial Neural Network", *2016 International Conference on Inventive Computation Technologies (ICICT)*, 2016.

[29]. S. Bhusri and S. Jain, "Analysis of breast lesions using laws' mask texture features," *2016 4th Int. Conf. Parallel, Distributed and Grid Computing(PDGC)*, pp. 56–60, 2016, doi: 10.1109/PDGC.2016.7913115.

[30]. S. Gharge, P. Singh, and Á. D. Á. Á. Neural, "Image Processing for Soybean Disease Classification and Severity Estimation," *Emerging Research in Computing, Information, Communication and Applications*, 2016, doi: 10.1007/978-81-322-2553-9.

[31]. S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, no. September, pp. 1–10, 2016, doi: 10.3389/fpls.2016.01419.

[32]. S. Prasad, S. K. Peddoju, and D. Ghosh, "Multi-resolution mobile vision system for plant leaf disease diagnosis," *Signal, Image and Video Processing*, vol. 10, no. 2, pp. 379–388, 2016, doi: 10.1007/s11760-015-0751-y.

[33]. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," *Computational Intelligence and Neuroscience*, vol. 2016, 2016, doi: 10.1155/2016/3289801.

[34]. S. Zhang and Z. Wang, "Cucumber disease recognition based on Global-Local Singular value decomposition," *Neurocomputing*, vol. 205, pp. 341–348, 2016, doi: 10.1016/j.neucom.2016.04.034.

[35]. X. E. Pantazi, D. Moshou, A. A. Tamouridou and S. Kasderidis, "Leaf Disease Recognition in Vine Plants Based on Local Binary Patterns and One Class Support Vector Machines," *International Federation for Information Processing*, pp. 319–327, 2016, doi: 10.1007/978-3-319-44944-9_27.

[36]. A. Johannes, A. Picon, A. A. Gila, J. Echazarra, S. R. Vaamonde, A. D. Navajas, and A. O. Barredoc, "Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case," *Computers and Electronics in Agriculture*, vol. 138, pp. 200–209, 2017, doi: 10.1016/j.compag.2017.04.013.

[37]. E. Hamuda, B. Mc Ginley, M. Glavin, and E. Jones, “Automatic crop detection under field conditions using the HSV colour space and morphological operations,” *Computers and Electronics in Agriculture*, vol. 133, pp. 97–107, 2017, doi: 10.1016/j.compag.2016.11.021.

[38]. G. Wang, Y. Sun, and J. Wang, “Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning,” *Computational Intelligence and Neuroscience*, vol. 2017, 2017, doi: 10.1155/2017/2917536.

[39]. H. Ali, M. I. Lali, M. Z. Nawaz, M. Sharif, and B. A. Saleem, “Symptom based automated detection of citrus diseases using color histogram and textural descriptors,” *Computers Electronics in Agriculture*, vol. 138, pp. 92–104, 2017, doi: 10.1016/j.compag.2017.04.008.

[40]. J. Ma, K. Du, L. Zhang, F. Zheng, J. Chu, and Z. Sun, “A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing,” *Computers and Electronics in Agriculture*, vol. 142, pp. 110–117, 2017, doi: 10.1016/j.compag.2017.08.023.

[41]. K. Dimililer and E. Kiani, “Application of back propagation neural networks on maize plant detection,” *Procedia Computer Science*, vol. 120, pp. 376–381, 2017, doi: 10.1016/j.procs.2017.11.253.

[42]. K. Kamal, R. Qayyum, S. Mathavan, and T. Zafar, “Wood defects classification using laws texture energy measures and supervised learning approach,” *Advanced Engineering Informatics*, vol. 34, pp. 125–135, 2017, doi: 10.1016/j.aei.2017.09.007.

[43]. M. Mehdipour Ghazi, B. Yanikoglu, and E. Aptoula, “Plant identification using deep neural networks via optimization of transfer learning parameters,” *Neurocomputing*, vol. 235, pp. 228–235, 2017, doi: 10.1016/j.neucom.2017.01.018.

[44]. M. Suresha, K. N. Shreekanth, and B. V. Thirumalesh, “Recognition of diseases in paddy leaves using knn classifier,” *2017 2nd International Conference for Convergence in Technology (I2CT)*, pp. 663–666, 2017, doi: 10.1109/I2CT.2017.8226213.

[45]. V. Singh and A. K. Misra, “Detection of plant leaf diseases using image segmentation and soft computing techniques,” *Information Processing in Agriculture*, vol. 4, no. 1, pp. 41–49, Mar. 2017, doi: 10.1016/j.inpa.2016.10.005.

[46]. X. Bai, X. Li, Z. Fu, X. Lv, and L. Zhang, “A fuzzy clustering segmentation method based on neighborhood grayscale information for defining cucumber leaf spot disease images,” *Computers and Electronics in Agriculture*, vol. 136, pp. 157–165, 2017, doi: 10.1016/j.compag.2017.03.004.

[47]. Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, “Identification of rice diseases using deep convolutional neural networks,” *Neurocomputing*, vol. 267, pp. 378–384, 2017, doi: 10.1016/j.neucom.2017.06.023.

[48]. Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, and K. Javed, “An automated detection and classification of citrus plant diseases using image processing techniques: A review,” *Computers and Electronics in Agriculture*, vol. 153, no. September 2017, pp. 12–32, 2018, doi: 10.1016/j.compag.2018.07.032.

[49]. C. S. Hlaing and S. M. Maung Zaw, “Tomato Plant Diseases Classification Using Statistical Texture Feature and Color Feature,” *17th IEEE/ACIS International Conference on Computer and Information Science (ICIS)*, no. 1, pp. 439–444, 2018, doi: 10.1109/ICIS.2018.8466483.

[50]. J. Ma, K. Du, F. Zheng, L. Zhang, Z. Gong, and Z. Sun, “A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network,” *Computers and Electronics in Agriculture*, vol. 154, no. August, pp. 18–24, 2018, doi: 10.1016/j.compag.2018.08.048.

[51]. K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, no. January, pp. 311–318, 2018, doi: 10.1016/j.compag.2018.01.009.

[52]. S. Kaur, S. Pandey, and S. Goel, “Semi-automatic leaf disease detection and classification system for soybean culture,” *IET Image Processing*, vol. 12, no. 6, pp. 1038–1048, 2018, doi: 10.1049/iet-ipr.2017.0822.

[53]. S. Zhang, H. Wang, W. Huang, and Z. You, “Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG,” *Optik (Stuttg).*, vol. 157, pp. 866–872, 2018, doi: 10.1016/j.ijleo.2017.11.190.

[54]. India economic survey 2018: “Farmers gain as agriculture mechanisation speeds up, but more R&D needed". The Financial Express. 29 January 2018. Retrieved 8 January 2019.

[55]. A. S. Deshapande, S. G. Giraddi, K. G. Karibasappa, and S. D. Desai, “Fungal disease detection in maize leaves using haar wavelet features,” *Information and Communication Technology for Intelligent Systems,* vol. 106. Springer Singapore, pp. 275-286, 2019.

[56]. G. Dhingra, V. Kumar, and H. D. Joshi, “A novel computer vision based neutrosophic approach for leaf disease identification and classification,” *Measurement,,* vol. 135, pp. 782–794, 2019, doi: 10.1016/j.measurement.2018.12.027.

[57]. G. Geetharamani and A. P. J., “Identification of plant leaf diseases using a nine-layer deep convolutional neural network,” *Computers and Electrical Engineering,* vol. 76, pp. 323–338, 2019, doi: 10.1016/j.compeleceng.2019.04.011.

[58]. J. Chen, Q. Liu, and L. Gao, “Visual tea leaf disease recognition using a convolutional neural network model,” *Symmetry (Basel),* vol. 11, no. 3, 2019, doi: 10.3390/sym11030343.

[59]. K. Singh, S. Kumar, and P. Kaur, “Support vector machine classifier based detection of fungal rust disease in Pea Plant (Pisam sativam),” *International Journal of Information Technology,* vol. 11, no. 3, pp. 485–492, 2019, doi: 10.1007/s41870-018-0134-z.

[60]. K. Zhang, Z. Xu, S. Dong, C. Cen, and Q. Wu, “Identification of peach leaf disease infected by Xanthomonas campestris with deep learning,” *Engineering in Agriculture, Environment and Food,* vol. 12, no. 4, pp. 388–396, 2019, doi: 10.1016/j.eaef.2019.05.001.

[61]. M. Sibiya and M. Sumbwanyambe, “An Algorithm for Severity Estimation of Plant Leaf Diseases by the Use of Colour Threshold Image Segmentation and Fuzzy Logic Inference: A Proposed Algorithm to Update a ‘Leaf Doctor’ Application,” *AgriEngineering,* vol. 1, no. 2. pp. 205–219, 2019, doi: 10.3390/agriengineering1020015.

[62]. N. Leena and K. K. Saju, “Classification of macronutrient deficiencies in maize plants using optimized multi class support vector machines,” *Engineering in Agriculture, Environment and Food*, vol. 12, no. 1, pp. 126–139, 2019, doi: 10.1016/j.eaef.2018.11.002.

[63]. S. Zhang, W. Huang, and C. Zhang, “Three-channel convolutional neural networks for vegetable leaf disease recognition,” *Cognitive Systems Research.*, vol. 53, pp. 31–41, 2019, doi: 10.1016/j.cogsys.2018.04.006.

[64]. X. E. Pantazi, D. Moshou, and A. A. Tamouridou, “Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers,” *Computers and Electronics in Agriculture*, vol. 156, pp. 96–104, 2019, doi: 10.1016/j.compag.2018.11.005.

[65]. A. Das, C. Mallick, and S. Dutta, “Deep Learning-Based Automated Feature Engineering for Rice Leaf Disease Prediction,” *Advances in Intelligent Systems and Computing*, vol. 1120. pp. 133–141, 2020, doi: 10.1007/978-981-15-2449-3_11.

[66]. A. Khamparia, G. Saini, D. Gupta, A. Khanna, S. Tiwari, and V. H. C. de Albuquerque, “Seasonal Crops Disease Prediction and Classification Using Deep Convolutional Encoder Network,” *Circuits, System and Signal Processing*, vol. 39, no. 2, pp. 818–836, 2020, doi: 10.1007/s00034-019-01041-0.

[67]. A. Zapolska, C. Kalaitzidis, E. Markakis, E. Ligoxigakis, and G. Koubouris, “Linear Discriminant Analysis of spectral measurements for discrimination between healthy and diseased trees of Olea europaea L. artificially infected by Fomitiporia mediterranea,” *International Journal of Remote Sensing*, vol. 41, no. 14, pp. 5388–5398, 2020, doi: 10.1080/01431161.2020.1731931.

[68]. B. Liu, Z. Ding, L. Tian, D. He, S. Li, and H. Wang, “Grape Leaf Disease Identification Using Improved Deep Convolutional Neural Networks,” *Frontiers in Plant Science*, vol. 11, no. July, pp. 1–14, 2020, doi: 10.3389/fpls.2020.01082.

[69]. E. Samatha, S. Chaturved, and C. Shailaja, “Plant Leaf Disease Detection and Classification using Texture Feature based Back Propagated Artificial Neural Network Classifier,” *International Journal of Advanced Science and Technology*, vol. 29, no. 8s, pp. 194–203, 2020.

[70]. J. Zhu, A. Wu, X. Wang, and H. Zhang, "Identification of grape diseases using image analysis and BP neural networks," *Multimedia Tools and Applications*, vol. 79, no. 21–22, pp. 14539–14551, 2020, doi: 10.1007/s11042-018-7092-0.

[71]. K. Ahmed, T. R. Shahidi, S. M. I. Alam and S. Momen, "Rice Leaf Disease Detection Using Machine Learning Techniques", *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI),* 2020, doi: 10.1109/STI47673.2019.9068096.

[72]. K. Khairnar and N. Goje, "Image processing based approach for diseases detection and diagnosis on cotton plant leaf," *Techno-Societal 2018 - Proceedings of the 2nd International Conference on Advanced Technologies for Societal Applications*. pp. 55–65, 2020, doi: 10.1007/978-3-030-16848-3_6.

[73]. M. T. Habib, A. Majumder, A. Z. M. Jakaria, M. Akter, M. S. Uddin, and F. Ahmed, "Machine vision based papaya disease recognition," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 3, pp. 300–309, 2020, doi: 10.1016/j.jksuci.2018.06.006.

[74]. N. Kaur and V. Devendran, "Novel plant leaf disease detection based on optimize segmentation and Law mask feature extraction with SVM classifier," *Materials Today: Proceedings*, 2020, doi: 10.1016/j.matpr.2020.10.901.

[75]. R. Kabir, S. Jahan, M. Rashedul Islam, N. Rahman, and M. R. Islam, "Discriminant feature extraction using disease segmentation for automatic leaf disease diagnosis," *Proceedings of the International Conference on Computing Advancements*, pp. 1-7, 2020, doi: 10.1145/3377049.3377100.

[76]. R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Applied Soft Computing,* vol. 86, p. 105933, 2020, doi: 10.1016/j.asoc.2019.105933.

[77]. R. Sreevallabhadev, "An improved machine learning algorithm for predicting blast disease in paddy crop," *Materials Today: Proceedings,* vol. 33, pp. 682–686, 2020, doi: 10.1016/j.matpr.2020.05.802.

[78]. S. Patidar, A. Pandey, B. A. Shirish, and A. Sriram, "Rice Plant Disease Detection and Classification Using Deep Residual Learning," *Communications in*

Computer and Information Science, vol. 1240 CCIS. pp. 278–293, 2020, doi: 10.1007/978-981-15-6315-7_23.

[79]. Y. A. Nanehkaran, D. Zhang, J. Chen, Y. Tian, and N. Al-Nabhan, “Recognition of plant leaf diseases based on computer vision,” *Journal of Ambient Intelligence and Humanized Computing*, 2020, doi: 10.1007/s12652-020-02505-x.

[80]. A. Gargade and S. Khandekar, “Custard Apple Leaf Parameter Analysis, Leaf Diseases, and Nutritional Deficiencies Detection Using Machine Learning,” *Lecture Notes in Electrical Engineering*, vol. 703. pp. 57–74, 2021, doi: 10.1007/978-981-15-8391-9_5.

[81]. A. Jose, S. Nandagopalan, C. Mouli, and V. Srinivas, “Artificial Intelligence Techniques for Agriculture Revolution : A Survey,” *Annals of the Romanian Society for Cell Biology*, vol. 25, no. 4, 2021.

[82]. D. Chauhan, R. Walia, C. Singh, M. Deivakani, M. Kumbhkar, and A. Professor, “Detection of Maize Disease Using Random Forest Classification Algorithm,” *Turkish Journal of Computer and Mathematics* Education, vol. 12, no. 09, 2021.

[83]. D. Mohapatra, J. Tripathy, and T. K. Patra, “Rice Disease Detection and Monitoring Using CNN and Naive Bayes Classification,” *Advances in Intelligent Systems and Computing*, vol. 1248. pp. 11–29, 2021, doi: 10.1007/978-981-15-7394-1_2.

[84]. G. Sambasivam and G. D. Opiyo, “A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks,” *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 27–34, 2021, doi: 10.1016/j.eij.2020.02.007.

[85]. H. Qi, Y. Liang, Q. Ding, and J. Zou, “Automatic identification of peanut-leaf diseases based on stack ensemble,” *Applied Sciences*, vol. 11, no. 4, pp. 1–15, Feb. 2021, doi: 10.3390/app11041950.

[86]. J. Kianat, M. A. Khan, M. Sharif, T. Akram, A. Rehman, and T. Saba, “A joint framework of feature reduction and robust feature selection for cucumber leaf diseases recognition,” *Optik*, vol. 240. 2021, doi: 10.1016/j.ijleo.2021.166566.

[87]. J. Shin, Y. K. Chang, B. Heung, T. Nguyen-Quang, G. W. Price, and A. Al-Mallahi, “A deep learning approach for RGB image-based powdery mildew

disease detection on strawberry leaves," *Computers and Electronics in Agriculture*, vol. 183. 2021, doi: 10.1016/j.compag.2021.106042.

[88]. M. Yogeshwari and G. Thailambal, "Automatic feature extraction and detection of plant leaf disease using GLCM features and convolutional neural networks," *Materials Today: Proceedings*. 2021, doi: 10.1016/j.matpr.2021.03.700.

[89]. N. Kaur, V. Devendran, and S. Verma, "Crop leaf disease classification identification based on ensemble," *Proceedings of Algorithms, Computing and Mathematics Conference 2021 (ACM 2021),* vol. 3010, pp. 110–118, 2021.

[90]. N. Kaur and V. Devendran, "Crop plant leaf disease prediction with law mask features using PCA and LDA with SVM," *International Conference on Challenges and Opportunities for Innovation in New Normal Scenario KSR Institute for Engineering and Technology,* Tiruchengode, Namakkal (Dt), Tamil Naduno. April, pp. 20–30, 2021.

[91]. N. Kaur and V. Devendran, "Ensemble Classification and Feature Extraction Based Plant Leaf Disease Recognition," *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO),* 2, pp. 1-4, 2021, doi: 10.1109/ICRITO51393.2021.9596456.

[92]. N. Kaur and V. Devendran, "Plant leaf disease detection using ensemble classification and feature extraction," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 11, pp. 2339–2352, 2021.

[93]. N. Kaur and V. Devendran, "Plant leaf disease detection using CNN and ensemble classification," *Design Engineering (Toronto)*, pp. 2372 - 2380, Jun. 2021.

[94]. N. Kaur and V. Devendran, "Plant Leaf Disease Diagnostic System Built on Feature Extraction and Ensemble Classification," *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO),* pp. 1-3, 2021, doi: 10.1109/ICRITO51393.2021.9596070.

[95]. P. Mekha and N. Teeyasuksaet, "Image Classification of Rice Leaf Diseases Using Random Forest Algorithm," *2021 Joint International Conference on Digital*

Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering, pp. 165–169, 2021, doi: 10.1109/ectidamtncon51128.2021.9425696.

[96]. Sutha P., Nandhu Kishore A. H., Jayanthi V. E., Periyanan A., Vahima P. "Plant Disease Detection Using Fuzzy Classification," *Annals of the Romanian Society for Cell Biology*, vol. 25, no. 04, 2021.

[97]. S. Mangal and P. Meshram, "Plant disease identification using deep learning classification model: CNN," *Journal of University of Shanghai for Science and Technology*, vol. 23, no. 01, 2021.

[98]. S. Mukhopadhyay, M. Paul, R. Pal, and D. De, "Tea leaf disease detection using multi-objective image segmentation," *Multimedia Tools and Applications*, vol. 80, no. 1, pp. 753–771, 2021, doi: 10.1007/s11042-020-09567-1.

[99]. S. Roy, R. Ray, S. R. Dash, and M. K. Giri, "Plant Disease Detection Using Machine Learning Tools With an Overview on Dimensionality Reduction," *Data Analysis Bioinformatics*, pp. 109–144, 2021, doi: 10.1002/9781119785620.ch5.

[100]. S. Sivagami and S. Mohanapriya, "Tomato leaf disease detection using image processing technique," *International Journal Agricultural Technology*, vol. 17, no. 3, pp. 1135–1146, 2021.

[101]. S. Xizhen, Z. Wei, G. Yiling, and Y. Shengyang, "Edge detection algorithm of plant leaf image based on improved Canny," *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*. pp. 342–345, 2021, doi: 10.1109/ICSP51882.2021.9408929.

[102]. Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using EfficientNet deep learning model," *Ecological Informatics*, vol. 61, Mar. 2021, doi: 10.1016/j.ecoinf.2020.101182.

[103]. V. K. Shrivastava and M. K. Pradhan, "Rice plant disease classification using color features: a machine learning paradigm," *Journal of Plant Pathology*, vol. 103, no. 1, pp. 17–26, Feb. 2021, doi: 10.1007/s42161-020-00683-3.

[104]. Y. Kurmi and S. Gangwar, "A leaf image localization based algorithm for different crops disease classification," *Information Processing in Agriculture*, pp. 1–19, 2021, doi: 10.1016/j.inpa.2021.03.001.

[105]. Y. Kurmi, S. Gangwar, D. Agrawal, S. Kumar, and H. S. Srivastava, “Leaf image analysis-based crop diseases classification,” *Signal, Image and Video Processing*, vol. 15, no. 3, pp. 589–597, Apr. 2021, doi: 10.1007/s11760-020-01780-7.

[106]. H. Qi, Y. Liang, Q. Ding, and J. Zou, “Automatic identification of peanut-leaf diseases based on stack ensemble,” *Applied Sciences*, vol. 11, no. 4, pp. 1–15, Feb. 2021, doi: 10.3390/app11041950.

[107]. “Watershed segmentation”, https://scikit-image.org/docs/dev/auto_examples/segmentation/plot_watershed.html

[108]. “Texture features”, https://cam-do.com/blogs/camdo-blog/macro-time-lapse-university-student-features-the-color-changing-procesfs-of-leaves

[109]. “S curve”, https://www.restore.ac.uk/srme/www/fac/soc/wie/research-new/srme/modules/mod4/4/index.html

[110]. “Convolutional layer”, https://www.embedded-vision.com

[111]. “Background study”, https://en.wikipedia.org/wiki/History_of_agriculture_in_the_Indian_subcontinent#:~:text=Indian%20agriculture%20began%20by%209000,techniques%20being%20developed%20for%20agriculture.s